AD-A118 197 NAVAL POSTGRADUATE SCHOOL MONTEREY CA F/G 5/1

TRIDENT SUBMARINE LOGISTICS DATA SYSTEM (LDS): A CASE STUDY IN --ETC(U)

MAR 82 G W HIZA

UNCLASSIFIED NL

1 of 2

AD
A118197

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

TRIDENT SUBMARINE LOGISTICS DATA SYSTEM (LDS): A CASE STUDY IN LIFE CYCLE MANAGEMENT AND BUDGETING

by

George W. Hiza

March 1982

Thesis Advisor: S. S. Liao

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO. AD-A118 197	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) TRIDENT Submarine Logistics Data System (LDS): A Case Study in Life Cycle Management and Budgeting		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis March 1982
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) George W. Hiza		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		12. REPORT DATE March 1982
		13. NUMBER OF PAGES 103
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

1. Life Cycle Management
2. Automated Information Systems Acquisition
3. Cost and Schedule Variance
4. Budgeting

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

As a method of controlling the rapidly rising costs and schedule delays plaguing software systems, Department of Defense (DOD) has implemented the concept of life cycle management for automated information systems (AIS). This thesis analyses the DOD life cycle management directives through the development of the TRIDENT Submarine Logistics Data System AIS. Specifically, it examines DOD software life cycle phasing and studies the cost

DD FORM 1473 1 JAN 73 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Block 20 (continued)

and schedule variance guidelines established by the life cycle management directives. This thesis points out an apparant need for clarifying the DOD budget guidelines and a refining of the life cycle documentation requirements.

Accession For	
NTIS GRA&I	☑
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A	

DD Form 1473
1 Jan 73
S/N 0102-014-6601

Approved for public release; distribution unlimited

TRIDENT Submarine Logistics Data System (LDS):
A Case Study in Life Cycle Management and Budgeting

by

George W. Hiza
Lieutenant, Supply Corps
United States Navy
B.S., State University of New York, 1973

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN MANAGEMENT

from the

NAVAL POSTGRADUATE SCHOOL
March 1982

Author: George W Hiza

Approved by:
Thesis Advisor

Second Reader

Chairman, Department of Administrative Sciences

Dean of Information and Policy Sciences

ABSTRACT

As a method of controlling the rapidly rising costs and schedule delays plaguing software systems, Department of Defense (DOD) has implemented the concept of life cycle management for automated information systems (AIS). This thesis analyses the DOD life cycle management directives through the development of the TRIDENT Submarine Logistics Data System AIS. Specifically, it examines DOD software life cycle phasing and studies the cost and schedule variance guidelines established by the life cycle management directives. This thesis points out an apparant need for clarifying the DOD budget guidelines and a refining of the life cycle documentation requirements.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

I. INTRODUCTION

A. COMPUTER SOFTWARE DEVELOPMENT PROBLEMS

Computer programs - generally called software packages - are instructions that tell computer systems what actions to take. As computer systems have become increasingly more sophisticated, attempts have been made to apply these systems to solving progressively more complex and intricate problems. Mismatches between the desired level of performance and the technical abilities to attain these levels of performance have become evident with the increasing complexity of software needs. The problems of writing and maintaining complex computer programs is causing computer software costs to outstrip hardware costs [Ref. 1]. A General Accounting Office (GAO) reports notes that by the mid-1980s over 90 percent of the cost of a computer system will be software costs [Ref. 2]. Figure 1 shows this relationship between hardware and software costs [Ref. 3].

The growing number of software project cost overruns, schedule slippages, user dissatisfaction and performance degradation in the recent past have created a growing appreciation for better management and control of personnel and dollar resources identified for these projects. A recent GAO survey indicated that government software development projects suffer

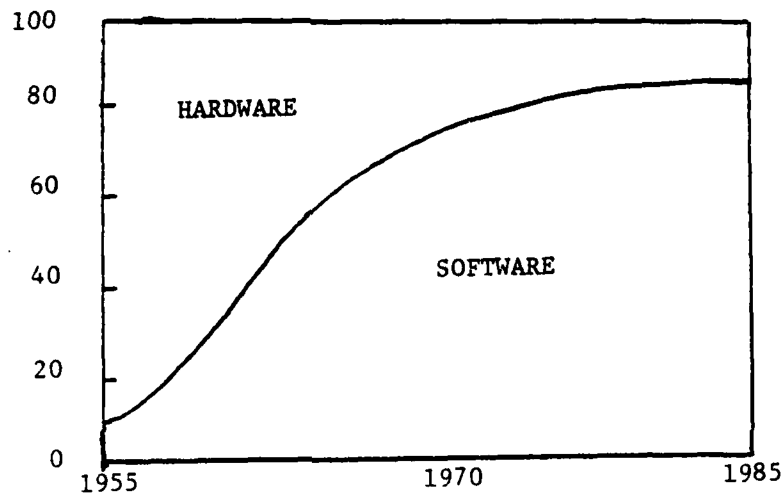

Figure 1. Hardware/Software Cost Trends

from these same problems [Ref. 4]. Figures 2, 3, and 4 show some of the survey questions and the responses to those questions.

B. DOD MANAGEMENT OF COMPUTER SOFTWARE PROJECTS

The Department of Defense (DOD) currently spends millions of dollars each year to develop, procure, and operate automated information systems (AIS). As defined by DOD Instruction 7920.1 entitled "Life Cycle Management of Automated Information Systems (AIS)", an AIS is:

> "... a collection of functional users and ADP personnel, procedures, and equipment (including ADPE) which is designed, built, operated, and maintained to collect, read, process, store, retrieve, and display information."

To be more specific, an AIS is a computer system, the management of which not only includes all the computer programs within the system, but also the computer hardware on which the software system will run.

In an effort to more efficiently control and manage its limited resources, DOD implemented life-cycle management procedures on all AIS with the exception of command and control and communication AIS with the promulgation of DOD Instruction 7920.1 in October, 1978. A new review and decision process for AIS was established by DOD Instruction 7920.2 entitled "Major Automated Information Systems Approval Process" also in October, 1978. Secretary of the Navy (SECNAV) Instruction 5231.1A entitled "Life Cycle Management of Automated Information Systems within the Department of the Navy" promulgated the

Response	Respondents	
	Number	Percentage
Very common	24	21.2
Fairly common	33	29.2
Not very common	29	25.7
Very rare	11	9.7
Never occurs	7	6.2
Don't know	9	8.0
Total	113	100.0

Figure 2. Software Development Has Dollar Overrun

Response	Respondents	
	Number	Percentage
Very common	34	30.1
Fairly common	36	31.9
Not very common	29	25.7
Very rare	9	8.0
Never occurs	2	1.8
Don't know	3	2.7
Total	113	a/ 100.0

a/ Does not add due to rounding

Figure 3. Software Development Has Calendar Overrun

Response	Respondents	
	Number	Percentage
Very common	10	8.8
Fairly common	39	34.5
Not very common	40	35.4
Very rare	15	13.3
Never occurs	7	6.2
Don't know	2	1.8
Total	113	100.0

Figure 4. The Delivered Software Must Be Corrected or Modified

policies and assigned the responsibilities for overall life-cycle management within the Department of the Navy in November, 1979.

These directives and instructions show a major change in the philosophy of managing computer software projects in the military. Prior to life-cycle management, DOD Directive 4105.55 entitled "Selection and Acquisition of Automated Data Processing Resources" and DOD Instruction 5100.40 entitled "Responsibility for the Administration of the DOD Automatic Data Processing Program" were the primary software development documents and concerned controlling the cost of acquiring software systems. These instructions asked the following questions: (1) Where are we? (2) Where do we want to be? (3) What specific steps are we going to take? (4) Who is responsible? (5) What resources are required?, and (6) Is the effort worth-while? [Ref. 5]. Still, the systems developed under them tended to cost much more than the original estimates and were delivered much later than expected.

Life cycle management considers the acquisition cost of the project plus operation, maintenance, and any other cost of an AIS project from program initiation throughout a stated life time or period of service for the project. Life cycle management is heavily weighted toward the developmental phases of the AIS. Decision points or milestones are interjected at specific times during the development process where the project is reviewed for accuracy in satisfying customer requirements

and compliance to cost and schedule constraints. Life cycle management stresses planning and is one of the primary methods of attempting to control spiralling software development costs and project delays in DOD.

Of particular importance to the transition to life cycle management is the requirement by DOD Instruction 7920.2 to create a Systems Decision Paper (SDP) for each major new AIS or major modification to an existing AIS and the maintenance of this document throughout the life of the AIS. The SDP will be the principal document for recording all the essential information on an AIS such as mission need, alternatives, cost/benefit analysis, budgets, future fiscal year funding needs, management plans, development plans, and test and evaluation plans and will be used by the Office of the Secretary of Defense (OSD) and DOD to support the decision making process regarding the AIS.

C. PROBLEMS FACING TRIDENT SUBMARINE LDS

The TRIDENT Submarine Logistics Data System (LDS) is a technically complex, totally integrated series of software programs that are being developed to support the operation of the TRIDENT submarine fleet. When implemented, the TRIDENT LDS will be the heart of a comprehensive coordinated logistics support network whose functioning will help the TRIDENT submarines attain stringent operational requirements.

In 1980, when the TRIDENT LDS was required to implement Life Cycle Management and the SDP reporting process, it had been under development for eight years, was approximately $19,000,000 over cost, and was only 40 percent complete.

The change to life cycle management created a number of problems for the various managers within the TRIDENT LSD. Of particular interest to this thesis are two questions which were raised regarding guidelines and constraints under which budgets were to be formulated and actual costs accumulated:

1. The separation of TRIDENT LDS costs into the categories of Design, Maintenance, and Management costs - Previous to implementing life cycle management, all costs attributable to the TRIDENT LDS were aggregated together into a single category or cost element within the TRIDENT Submarine Project. The categories of Design, Maintenance, and Management stemmed primarily from attempting to define the acquisition/development approval authority thresholds for the TRIDENT LDS and those functions which constituted development costs and maintenance costs.
2. Application of the budgeting cost and schedule variances established by the life cycle management instructions and directives - Estimating the costs and time required to complete software development projects tends to be ambiguous and difficult. The precariousness of these estimates escalates dramatically as the timing,

technology, and complexity demanded from the projects increases. The TRIDENT LDS AIS does not appear to fit into the developmental mold described in the life cycle management instructions and the cost and time constraints seem to impose an artificially firm budget and schedule to portions of the project that are to be developed three, four, or more years in the future and whose functional capabilities have not been determined.

D. THESIS OBJECTIVES AND METHODOLOGY

This thesis is aimed at investigating software development processes in order to provide a definition through which TRIDENT LDS functions and costs may be designated into the appropriate Design, Maintenance, or Management category and examining budgeting and budget guidelines so that application of the cost and schedule variances may be determined.

Additionally, a comparison is made between the manner in which the TRIDENT LDS project is being developed, guidelines provided by DOD, and 'theoretical' development phases for the purpose of highlighting any procedural or conceptual differences which could have been bearing on budgeting and recommending changes to the process.

In conducting the investigation a search of journals, periodicals, books, and government documents was accomplished. This was done to develop the author's level of knowledge from which evaluation of the TRIDENT LDS could be made. Further,

field trips were made to the TRIDENT LDS ADP Manager Fleet Material Support Office (FMSO 96T), Mechanicsburg, PA so that current methodology used for budgeting and software development in the TRIDENT LDS could be studied. It is on these research efforts and the information obtained that the weaknesses are highlighted, conclusions drawn, and recommendations based.

E. ORGANIZATION OF THESIS

Chapter II discusses the development of the TRIDENT submarine and the basic concept of Integrated Logistic Support (ILS) for it, describes the TRIDENT LDS program, and outlines the TRIDENT LDS Systems Decision Paper (SDP). Chapter III compares software life cycle phases as described in management information system books and industrial situations with the DOD life cycle phases and the development of the TRIDENT LDS. Differences are noted and a method for phasing software development presented. Chapter IV addresses budget processes, discusses the division of software development function and costs into Design, Maintenance, and Management categories, and projects some interpretations in applying the variance constraints established by DOD Directive 7920.1 and SECNAV Instruction 5231.1A. Finally, Chapter V offers a summary, conclusions, and recommendations for areas of future study.

II. TRIDENT SUBMARINE LOGISTIC SUPPORT

A. DEVELOPMENT OF THE TRIDENT SUBMARINE LOGISTICS CONCEPT

The TRIDENT submarines scheduled for deployment during the 1980s are intended to become the primary sea based weapons system in the United States strategic deterrent forces [Ref. 6]. Currently there are seven TRIDENT submarines under contract for construction, one TRIDENT contract scheduled for approval during fiscal year 1981, and procurement of an additional eighteen TRIDENTs identified in future fiscal year budget submissions. At present, the goal is to have two squadrons of TRIDENT submarines each with ten operational ships. Although the projected number of TRIDENT submarines is significantly less than the size of the current United States Polaris/Poseidon fleet, a decision was made that the TRIDENT fleet would have a higher on-line availability than the Polaris/Poseidon fleet [Ref. 6]. In order to achieve higher levels of on-line availability, the on-line capability of each TRIDENT submarine had to be increased. Chief of Naval Operation identifies an operating cycle for TRIDENT submarines which requires longer patrol periods, shorter refit periods, a shorter and less frequent shipyard overhaul periods. TRIDENT submarines are to operate on a 70-day patrol/18-day refit cycle for a period of not less than nine years between scheduled 12 month shipyard overhaul periods.

The requirement for increasing on-line availability significantly affected the development of the overall TRIDENT project in a number of areas:

1. The design of the submarine was affected by attempting to increase equipment and component maintenance and reliability factors and by increasing accessibility to equipment in order to facilitate equipment repair or replacement.

2. A maintenance strategy was developed which called for the planning and scheduling of all maintenance actions at all levels for all patrols and refits from initial deployment of each ship through scheduled shipyard overhauls. This maintenance program includes all maintenance to be accomploshed on board each ship each patrol by ship's force personnel; coordination of the Intermediate Maintenance Activity (IMA) for maintenance it will perform each refit cycle; augmentation of IMA maintenance by periodic planned replacement of equipment prior to their expected failure time; and coordination of depot level maintenance for repair of items removed from the submarines which require depot level maintenance action.

3. All logistic requirements — repair parts, spares, tools, technical documentation, industrial facilities, etc. — are to be planned and controlled.

4. All data regarding equipment configuration and maintenance practices is to be continuously accumulated and updated in order to keep logistic support current with the equipment configuration.

Coupling these requirements to the requirement for a logistic information capability for the TRIDENT submarine as identified in OPNAV Instruction 4000.82 entitled "Logistics Support of the TRIDENT System" generated the need for a high intensity, meticulously managed Integrated Logistics Support (ILS) program. A program with this type of logistic information capability is not currently available to the Navy [Ref. 7].

B. INTEGRATED LOGISTICS SUPPORT (ILS)

ILS as described by Chief of Naval Material (NAVMAT) Instruction 4000.20B entitled "Integrated Logistics Support (ILS) Planning Policy" and DOD Directive 4100.35 entitled "Integrated logistics support planning guide for DOD systems and equipment" is:

> "A composite of all the support considerations necessary to assure the effective and economical support of systems/ equipments for their life cycle. It is an integral part of system/equipment acquisition and operation and is characterized by harmony and coherence among all logistic elements."

ILS is based on detailed analysis of all interaction and interdependency of equipment/component/system hardware design, development and performance specifications, and known or projected support requirements. The ILS process also identifies the resources necessary to support any operation and maintenance functions and strives for reducing the support burden placed on operating forces [Ref. 8]. The principal elements related to the ILS concept are listed in Appendix A.

The ILS concept is extremely important not only because it aids earlier identification of life-cycle costs and can help reduce total project costs but also because without adequate support, equipment and systems may not be able to meet expected operational capabilities. Systems which cannot operate satisfactorily in prescribed environments for a specified length of time and, when failed, cannot be restored to service within a specified length of time will not satisfy operational requirements [Ref. 9]. Additionally, the availability of items needed for system operation and maintenance such as test equipment, trained personnel, and repair parts will impact satisfying operational requirements.

An ILS plan for the TRIDENT Submarine System has been promulgated by the TRIDENT Systems Project Manager (Chief of Naval Material PM-2). This plan assigns the responsibility for planning, coordinating, developing, and integrating all logistic elements required to support TRIDENT submarines from acquisition through operation into a TRIDENT Logistic Support System. This Logistic Support System includes [Ref. 10]:

1. A refit facility and a training facility located at Bangor, Washington, which are dedicated to providing maintenance, refit services, supply support, and crew training for TRIDENT submarines.

2. A TRIDENT support organization in Mechanicsburg, PA whose responsibility is to provide technical and management support for TRIDENT logistic requirements.

3. Logistic Element Managers (LEMs) whose responsibility is to identify, acquire, and manage logistic resources applicable to their specific equipment.

4. A TRIDENT logistic information system that can coordinate and perform all the logistic functions required for a complete ILS system.

This logistics information system — the TRIDENT Logistics Data System (LDS) is discussed in the following section.

C. TRIDENT LOGISTIC DATA SYSTEM (LDS)

The TRIDENT LDS currently under development is a key element in implementing the total ILS concept for the TRIDENT Submarine System. The TRIDENT LDS is a shore based dedicated AIS having the objective:

> "... to provide an integrated information system necessary to support the intensified level of maintenance and logistics support required for TRIDENT submarines to achieve their high level of operational availability [Ref. 11]."

Its development and degree of success will be important to other DOD activities and to the development of future ILS projects because the TRIDENT LDS is the first time that an attempt has been made to implement the ILS concept for an entire weapons system. Additionally, it is being developed in such a manner as to interface with other standard Navy information systems such as the Fitting Out Management Information System (FOMIS), the Weapons System File (WSF), the Navy Maintenance Material Management (3M) System, and the Uniform Automated Data Processing System (UADPS) [Ref. 12].

The TRIDENT LDS developed through three phases since its inception. Development began in 1972-1973 prior to preparation of detailed Requirements Statements (RS) describing user functions that had to be satisfied by the data system. Initially the TRIDENT LDS was conceived as a central computer system located at the TRIDENT Support Activity in Mechanicsburg, PA that was to be linked to remote terminals located at the TRIDENT Refit Facility (TRIREFFAC) in Bangor, WA. By the time the formal RSs were created in 1975-1976, the centralized computer idea was changed and the decision made to provide computer capabilities at the TRIREFFAC in order to facilitate scheduling of maintenance action to be performed during the short, time-sensitive refit periods. Also during this period plans were developed which would resolve some incompatibilities that had emerged between operational data systems and allow them to interface with each other and with the TRIDENT LDS. The TRIDENT LDS began its third phase of development in 1977 when systems requirements were refined, software programming started, and hardware procured.

During these three phases of development, an LDS project completion date of September 30, 1980 had been established. By December, 1978, a decision was reached that the TRIDENT LDS project would not achieve its scheduled completion date and that projected cost of the project would be in excess of the 25 percent cost growth allowed by the Automated Data System Development Plan (ADS Plan). As required by the ADS Plan when

time and cost estimates can not be met within prescribed limits, a TRIDENT LDS project review was conducted and revised cost estimates and time schedules developed. These revisions were approved but along with the approval was the requirement to implement life-cycle management and the SDP process as set forth in DOD Directive 7920.1, DOD Instruction 7920.2, and SECNAV Instruction 5231.1A.

The TRIDENT LDS is organized into five major information areas which provide TRIDENT LDS users with data necessary to provide logistic support within that functional area. A sixth LDS branch creates the operating environment needed in order to operate the programs on the LDS hardware. Figure 5 shows the TRIDENT LDS tree [Ref. 13] and Appendix B summarizes the functions within each major LDS branch.

The development of the TRIDENT LDS has been segmented into five phases or revisions. Each revision represents a level of effort needed to implement a specific enhanced operational capability to the TRIDENT Submarine System. It is to these revisions that budgeting and cost accumulation are to be directed. Figure 6 is a matrix that shows the interrelationships between LDS revision numbers, the major system or branch, the SDP AIS milestone, and projected completion dates of each SDP milestone within a specific TRIDENT LDS revision [Ref. 14]. The SDP AIS milestones are explained in Chapter III. The 'Release' column on Figure 6 represents a major branch update/ verification to ensure that the branch will continue to operate

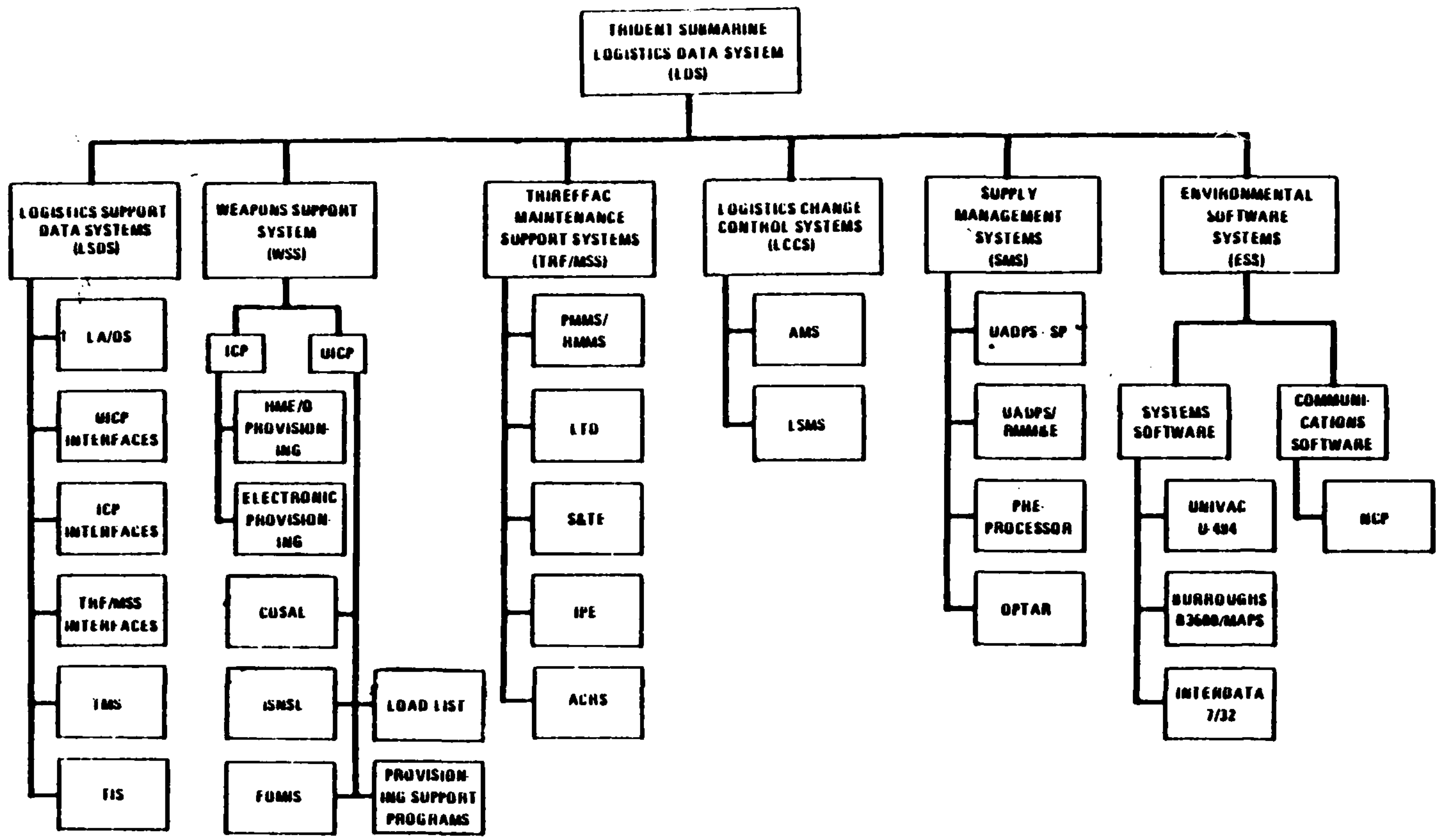

Figure 5. TRIDENT LDS Applications Systems

LDS REV	SYSTEM	RELEASE	MILESTONES 0	I	II	III
0	LSDS	1				
	WSS	1				
	TRF/MSS	N/A				
	SMS	N/A	COMPLETE	COMPLETE	COMPLETE	COMPLETE
	LCCS	N/A				
	ESS	1				
	HARDWARE	1				
1	LSDS	2				
	WSS	1				
	TRF/MSS	1				
	SMS	1	COMPLETE	COMPLETE	SEPTEMBER 1980	AUGUST 1981
	LCCS	N/A				
	ESS	2				
	HARDWARE	2				
2	LSDS	3				
	WSS	1				
	TRF/MSS	2				
	SMS	2	COMPLETE	SEPTEMBER 1980	SEPTEMBER 1981	SEPTEMBER 1983
	LCCS	1				
	ESS	3				
	HARDWARE	2				
3	LSDS	4				
	WSS	1				
	TRF/MSS	3				
	SMS	3	COMPLETE	JANUARY 1981	JANUARY 1982	JANUARY 1985
	LCCS	1				
	ESS	4				
	HARDWARE	3				
4	LSDS	5				
	WSS	1				
	TRF/MSS	4				
	SMS	4	COMPLETE	JULY 1983	JULY 1984	JULY 1987
	LCCS	2				
	ESS	5				
	HARDWARE	4				

Figure 6. TRIDENT LDS Milestone Status

correctly with another branch which may have been changed as the result of a TRIDENT LDS revision [Ref. 15].

D. TRIDENT LDS SYSTEM DECISION PAPER (SDP)

DOD Instruction 7920.2 states that:

> "The successful management of an AIS requires that the combined and integrated efforts of functional, ADP, and telecommunications organization and personnel. The SDP process provides for appropriate policy level involvement in key decisions during the life cycle of each major AIS."

An SDP is projected to be a living document in existence throughout the life cycle of an AIS. Once the Mission Element Needs Statement (MENS) describing a specific mission deficiency and justifying the need to seek alternate methods of solving the deficiency has been approved by the Secretary of Defense (for major AIS), an SDP is prepared by the AIS Project Manager for use in DOD and OSD decisions regarding continued development of the AIS. If approved by the OSD, the SDP is returned to the applicable DOD activity for further work on the AIS. Figure 7 shows the approval and management organization of the TRIDENT LDS [Ref. 16].

The SDP is based on the four specific AIS SDP milestones and related status and the five developmental phases for an AIS described in DOD Directive 7920.1. When all tasks required to progress from a previous milestone are completed, the SDP is updated and resubmitted to the OSD for review and approval to continue to the next phase of developing the AIS. During this OSD review process, any conflicts such as between

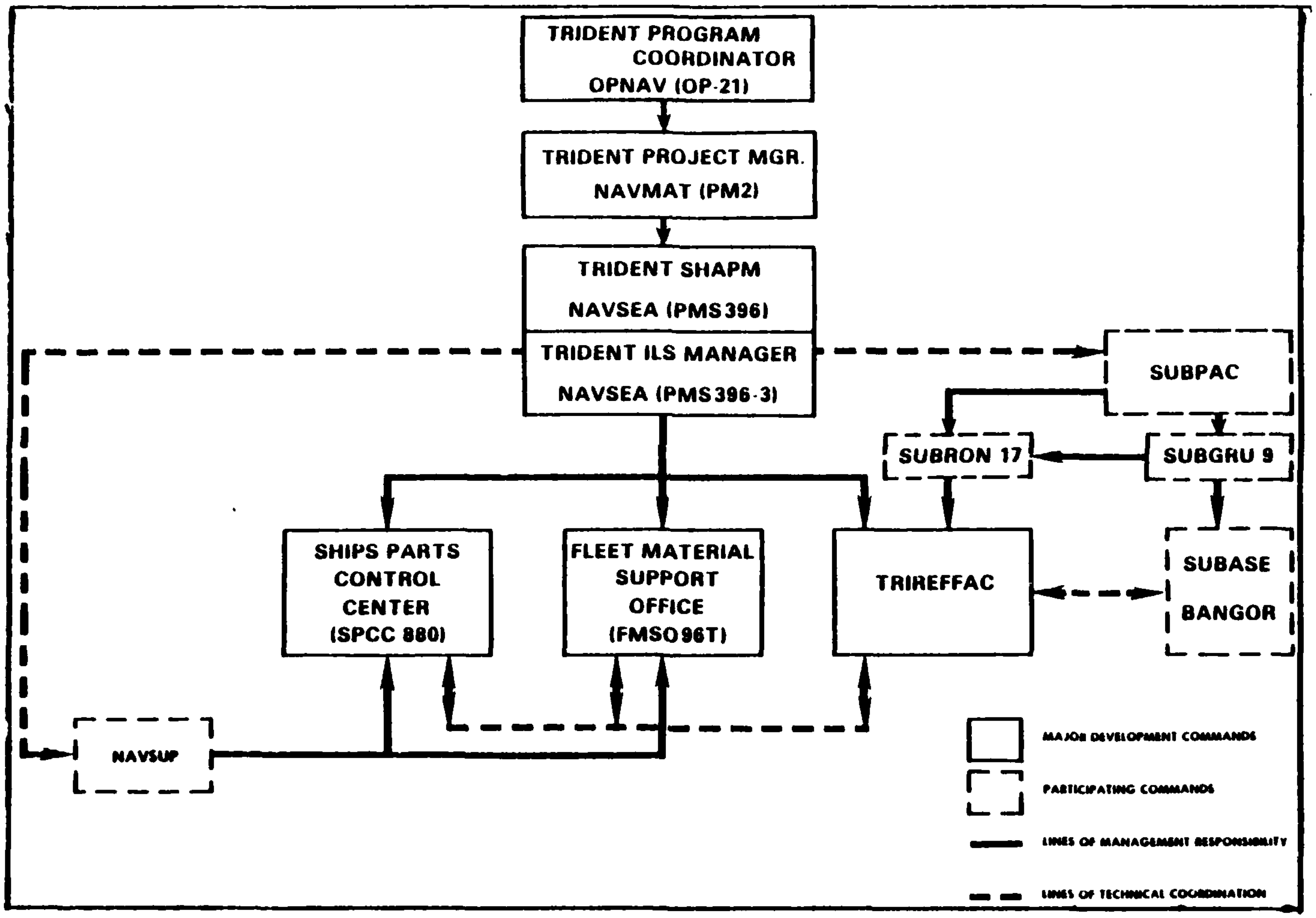

Figure 7. TRIDENT LDS Development Organization

projected cost and schedule goals or guidance as given by the OSD and actual direction taken by the SDP is documented and the ADP endorsed to reflect the OSD recommendations and decitions. As endorsed, the SDP is returned to the applicable DOD activity and, if the SDP has been approved, development of the AIS continued. Because of the tremendous amount of work that had been accomplished on the TRIDENT LDS under the ADS Plan and the effort involved in transitioning to the SDP process, development of the TRIDENT LDS continues bu the formal SDP has yet to be approved by OSD.

As required by DOD Instruction 7920.2, the TRIDENT SDP contains:

1. The MENS and a user requirements summary identifying the basic user requirements to be satisfied by the TRIDENT LDS.

2. The project plan including the description of the system, the plan by which the system will be managed and by whom, and the plan describing the manner and methodology of developing the system.

3. The acquisition strategy concerning TRIDENT LDS hardware, software, and supporting telecommunications requirements.

4. A logistics and training plan for the system.

5. Resources requirements including a Cost/Benefit Analysis (CBA) of alternatives considered.

6. A test and evaluation plan for conducting hardware and software tests, system effectiveness reviews, and acceptance tests.

III. AIS SOFTWARE LIFE CYCLE DEVELOPMENT

This chapter discusses the various phases that theoretical software systems pass through during their life cycle and the Automated Information System (AIS) life cycle phases described by Department of Defense (DOD) Directive 7920.1. The development of the TRIDENT Logistics Data System (LDS) is then presented and differences in the way it is being developed noted. A background is established in this chapter that helps highlight weaknesses in the DOD life cycle phasing which could cause budgeting problems. It also assists in separating software functions and related costs into the Design, Maintenance, and Management budget and cost accumulation categories addressed in Chapter IV.

A. THEORETICAL SOFTWARE LIFE CYCLE DEVELOPMENT

A computer based information system has a life cycle that is analagous to the life cycle of a living organism. Whether it is called a life cycle, a development cycle, or an implementation cycle, they mean essentially the same thing. [Ref. 17] A software system begins its life cycle when a need to improve information processing procedures is stimulated and ends its life cycle with disposal when its existance no longer serves the need or the need is no longer present/has been superceded by a higher priority need. Depending upon the degree to which one desires to separate the activities which

take place within a software life cycle, there are usually from four life cycle phases [Ref. 18] to ten life cycle phases [Ref. 19]. In general, a software life cycle can be separated into the following phases: (1) Analysis Phase, (2) Feasibility Study Phase, (3) Design Phase, (4) Program Development and Test Phase, (5) Evaluation Phase, and (6) Installation and Operation Phase. While covering the entire life cycle of the software system, these phases concentrate on the logical, accurate creation of the system and stress its planning.

1. Analysis Phase

This phase begins with the need for a new product and the acknowledgement of this need by the orgainzation's management. Concentration of what the need or problem is and not how it is to be solved is made during this phase. Ths proposed software user/customer and problem environment are identified, the role that the proposed product will play in satisfying the need is determined, and current capabilities/ state of art defined. These aspects are combined into a "Requirements Statement" (RS) or problem specification describing in detail the goals and objectives of the proposed system, the capabilities to be included in and excluded from the system, performance/processing specifications such as input rates, display times, file/record maintenance, output requirements and reports, interface requirements, and timing constraints.

2. Feasibility Study Phase

The feasibility study phase is sometimes considered an extension of the analysis phase, only more technically oriented. Existing procedures are examined in order to determine if any existing files, programs, and applications can be used or modified to help solve the need and which areas of the proposed system must be designed from scratch. Alternate methods of solving the problem are developed and each alternative along with the specific problem are studied to determine the feasibility of developing it. Feasibility is broken down into "operational feasibility" and "economic feasibility". Operational feasibility looks at whether or not the product will work performing its specific requirements in an expeditious manner - can input data be collected, erros corrected, and the system run on a set schedule? Economic feasibility looks at developing the product for a reasonable cost and the estimated cost effectiveness of the system when in operation [Ref. 20]. Estimates of potential costs, time, and effort must be made for developing the product as well as projections made for operating the product. Table I lists some project selection criteria that should be evaluated during the decision making process [Ref. 21]. The selection of a single alternative to pursue leads into the next life cycle phase.

TABLE I

SOME POTENTIAL CRITERIA FOR EVALUATING ALTERNATIVES IN PROJECT SELECTION

- Tangible and intangible benefits
  - User satisfaction
  - Percentage of needs met
  - Maximum potential of application
  - Costs of development
  - Costs of operations
- Timing of costs
- Timing of benefits
- Impact on existing operations
- Development time
- Time to implement
- Manpower required
  - Analyst
  - Programmer
  - User
- Probability of success
- Probability of meeting estimates
- New equipment required
- Priority of function

3. Design Phase

While some preliminary drafting and sketching of design ideas is accomplished during the feasibility study phase in order to support the decisions made, it is during the design phase that the systems analysts get down to designing a software structure that satisfies the user's requirements detailed in the RS. This is usually accomplished through successive iterations of the product until it is realistic [Ref. 22]. The principal product of the design phase is the Design Specification which describes how the planned system will be structured in order to satisfy all the requirements of the RS [Ref. 23]. The design specification is the foundation or baseline for all program implementations. It includes [Ref. 24]:

- -- a brief narrative and diagrams providing an overview of the entire system
- -- the standards and conventions or rules adopted for use in the programs such as flow charting standards; naming standards; interface of communication standards between program modules, components, operations, etc.; and coding standards to be used during the programming phase
- -- system file design and layout including subdivisions, files, field length, identifying characters, and file relationships and links

-- data flow diagrams describing all data transactions in the system to provide understanding of data paths and major events in the operating system.

Table II contains a list of items which should be included in the design specification [Ref. 25]. Additionally, during the design phase, the test specifications describing the project and the implementation plan detailing all measureable milestones, assignments, resources, and schedules are produced [Ref. 23]. At the end of the design phase the project is almost at a point of no return [Ref. 26]. Major amounts of resources are about to be committed and the design had better be correct. A detailed review of the design specification is conducted and, if approved, programming started.

4. Program Development and Testing Phase

During this phase the actual work of building the software program takes place. The internal design of the program is developed, programs are coded, flow charts and other system's documentation created and maintained, and testing and program debugging accomplished. Unit tests or individual tests of low level modules are performed initially by the programming teams. As these low level modules are made to perform in accordance with the user's requirements, they are integrated or strung together to create larger and larger portions of the overall project. These integrated groups are tested and debugged until the complete system has

TABLE II

DETAILED SYSTEMS DESIGN SPECIFICATIONS

- Output
  - Destination and use
  - Medium
  - Reports (samples)
  - Frequency
- Input
  - Source
  - Medium
  - Document (sample)
  - Fields
  - Estimated volume
- Files
  - Medium
  - Contents
  - Record format, field names
  - File structure (linkages, directories)
  - Estimated file size
  - Updating frequency
- Processing
  - System flow
  - Program specifications
    - Input
    - Output
    - Errors
    - Design decisions
    - Modules
    - Processing
- Conversion programs
  - Input
  - Output
  - Errors
  - Design decisions
  - Modules
  - Processing
- Manual procedures
- Error control
  - Input error conditions
  - Processing errors
  - File integrity
  - Output errors
  - Backup
  - Security
- Work plan
  - Program schedule, milestones
  - Time estimates
  - Personnel required assignments

been put together and progressively tested. Figure 8 shows the hierarchy of software project testing [Ref. 27].

5. Evaluation Phase

This phase acts as a buffer zone between the integrated testing performed by the programmers in the previous phase and the start of live use of the product. Its main objective is to subject the programmers' products to a thorough set of tests neither designed nor executed by them and run in an environment that as closely resembles the actual environment as possible [Ref. 28]. Test data used should include as many different system's conditions as possible and a sample of each type of transaction which will occur during operations. Illegal transactions, incorrect data entries, improperly coded data, as well as correct data transactions should be included in the test data to be sure that the programs can operate correctly and have adequate error checking and editing features built into them. Subsequent to the systems testing, the software product is presented to the user for acceptance testing. The acceptance test criteria are the conditions that the product must satisfy before the user finally accepts the product and agrees that it is free of defects and satisfies the specifications of the RS [Ref. 29]. Additionally during this phase all the software reference documentation is made available to and used to help the user on the system. This documentation includes program instructions, design documentation, flow charts, user manuals,

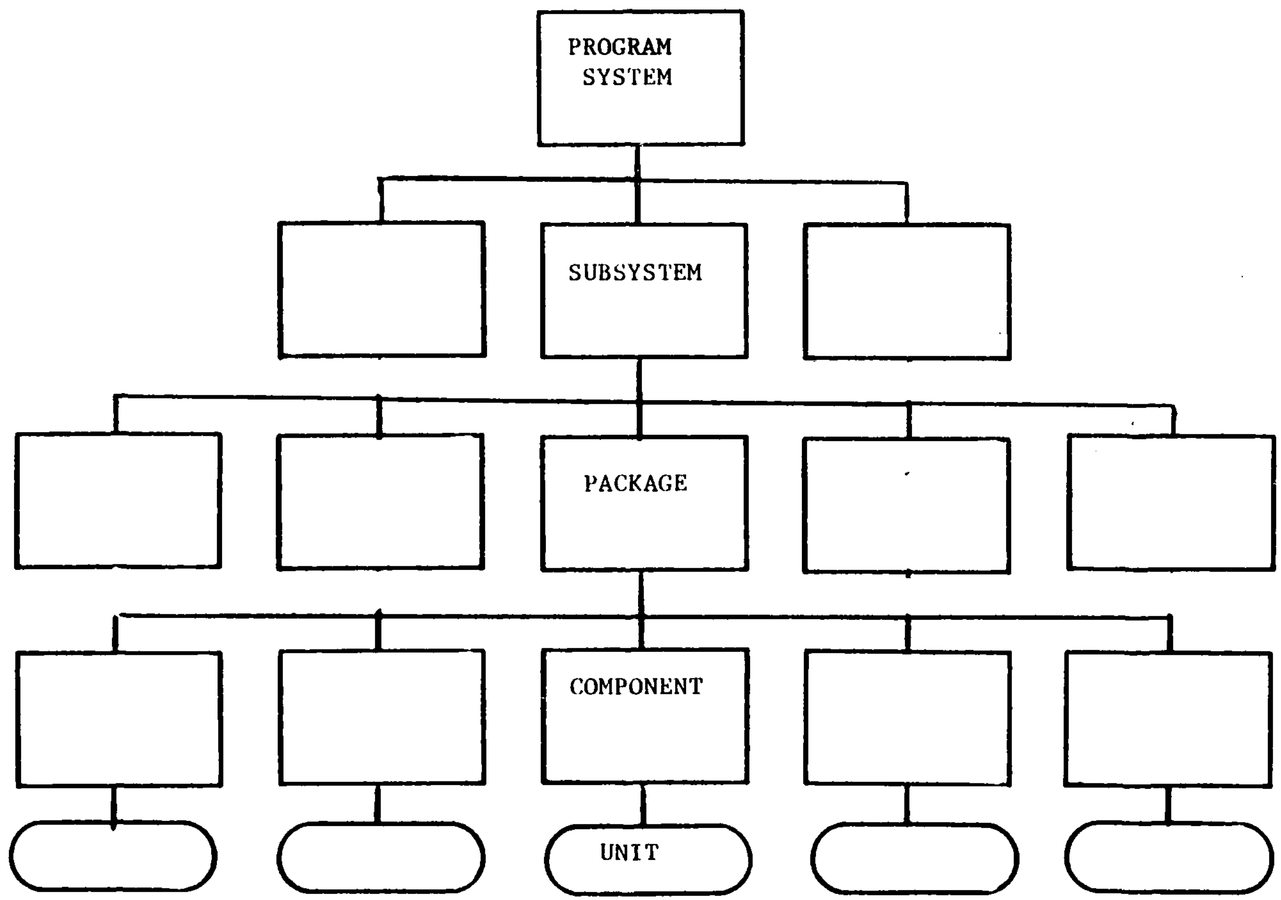

Figure 8. Testing Hierarchy.

operator manuals, maintenance manuals, error list conditions, and any other documentation that will make the system easier to understand and operate.

6. Installation and Operation Phase

For the most part acceptance testing is conducted on the user's equipment but for many systems acceptance testing is conditional and is followed up by installing the new system at the user's operational site and then testing it for proper operation. The new software system is generally replacing some other type of system – manual, automated, or a combination of both – and, therefore, the user's operations need to be converted over to the new system after the operational site testing has been completed. At this stage, the software product generally transitions into the operational phase when the system is in active and productive use. The operational phase includes activities such as continued training, tuning and maintaining the system, and possibly system enhancement and lasts until the product is withdrawn from active service and disposed of.

B. DOD AIS LIFE CYCLE PHASES AND SDP MILESTONES

As with theoretical software systems, DOD has developed a life cycle plan for its automated information systems through which their development and continued operation is managed. DOD Directive 7920.1 separates the life cycle of an AIS into five broad phases: (1) Mission Analysis/Project

Initiation, (2) Concept Development, (3) Definition/Design, (4) System Development, and (5) Deployment/Operations. It also establishes four milestones which help control and validate the development of the AIS. Prior to approval to proceed from one milestone to the next, specific assigned tasks must be completed, policy decisions made, and resource requirements (time and cost) confirmed. At each milestone, a decision is made to approve continued development of the AIS, establish corrective action in order to get the project back on track, or discontinue development action.

1. <u>Mission Analysis/Project Initiation</u>

This phase of AIS development identifies and validates a specific mission need and the deficiencies which prevent the successful accomplishment of the mission and presents a recommendation for analysing various ways by which the mission need may be satisfied. The Mission Element Needs Statement (MENS) is the method through which this is accomplished. The MENS describes a mission need in terms of the job to be done and the expected mission results. It describes the mission deficiency or non-performance and the impact on the ability to accomplish the mission without the new capability. Constraints such as operational and logistic limitations; interface with existing AIS; timing of need; interservice, intraservice, and interoperability requirements; and resource limitations are also identified in the MENS. This phase of AIS development ends with the approval of the MENS and

authorizes the analysis and development of alternate methods by which to resolve the deficiencies.

SDP Milestone 0 represents the termination of the Mission Analysis/Project Initiation Phase.

2. Concept Development

During the second phase of AIS development alternate methods of accomplishing the mission need identified in the MENS are developed and evaluated. These alternatives are so described as to reflect the various state of the art and technology bases available to solve the deficiency satisfactorily. One or more of these alternatives are designated for further evaluation. Modeling and simulation are used to establish feasible conceptual baselines for future research. Interface between ADP, telecommunications, logistics, and other elements plus comparison between in-house and contractor performance are introduced to the evaluation process during this phase. The significant tasks and policies required during the development phase are:

-- mission need is reaffirmed as necessary

-- project manager and staff assigned

-- functional objectives prioritized

-- development of detailed functional descriptions including inputs, processes, outputs, and interfaces

-- estimated resource requirements are bounded by established contraints

-- preliminary project plans are established which include concepts for training, operation, logistic support, and organizational relationships

-- alternatives have considered the use of existing hardware and software systems

-- risk and uncertainty areas are identified and included in planning and evaluation

-- preliminary test and evaluation plans are established

Demonstration of alternatives or approval to proceed directly to the Definition and Design Phase completes this phase and is designated as SDP Milestone 1.

3. Definition/Design

The system/subsystem specifications and functional operational requirements are fully defined during this phase of AIS development. Hardware, software, and data base specifications are developed. A detailed description of the functions to be supported by automation is created and specific objectives in terms of performance measuring are established and developed during this phase. Feasibility studies and economic analysis are prepared in support of these objectives plus training requirements, schedules, and projected costs.

SDP Milestone 2 completes this phase of development and represents the approval to fully develop the AIS.

4. Systems Development

During the fourth phase of the AIS life cycle, the total AIS is developed, integrated, tested, and evaluated. Computer programs, all data bases, and all system support documentation — users manuals, maintenance manuals, operators manuals — are developed and published. Interrelationships and interoperability with other AIS is included in the system development. System management and development plans and test and acceptance plans are defined during this phase and the project held to within the constraints of the resources allocated to it. Life cycle schedules and cost estimates are validated realistic, acceptable, and supportive of cost effective operations. Hardware and software are field tested using actual functional data and certified for satisfying system requirements.

SDP Milestone 3 represents completion of this phase and is the approval to deploy and operate the AIS.

5. Deployment and Operation

The purpose of this last phase of the AIS life cycle is to implement the approved operational plan, continue operations, and budget for continued operations and any modifications/changes throughout the useful life of the system. Training and resource requirements are to be kept current, operational efficiency and effectiveness periodically reevaluated, and major changes approved using the SDP process.

No SDP Milestone exists for this last phase of AIS life cycle management.

C. TRIDENT LDS DEVELOPMENT

In this section the manner in which the TRIDENT LSD is being developed is presented. Additionally, where apparant differences exist between the theoretical life cycle phases and the DOD life cycle phases, these differences are discussed.

1. Development by Revision

The TRIDENT LDS has been separated into the basic DOD life cycle phases and assigned SDP developmental milestones in accordance with the prevailing instructions. As stated in Chapter II, the TRIDENT LDS capability also has been separated into stages or revisions (see Figure 6). Revision of the TRIDENT LDS represents the inital system capability organizing the Integrated Logistics Support (ILS) data and supporting acquisition of the lead TRIDENT submarine. Revision 0 has been completed for all LDS revisions. Revision 1 provides system capability to support the first TRIDENT submarine refit at the TRIREFFAC and incorporates initial TRF/MSS and SMS capabilities, the operational hardware at the TRIREFFAC, and a system test bed configuration at the TRIDENT support Activities in Mechanicsburg, PA. Revisions 2 and 3 provide enhanced system operational capabilities by incorporating initial LCCS configuration change control tracking and feed-back systems and reorientating the LA/OS module of the LSDS branch from an acquisition perspective to an operational perspective respectively. Revision 4 completes the LCCS and

LSDS branch capabilities and incorporates system hardware improvements for the TRIREFFAC and the Systems Command Headquarters. These changes complete the multiship, overhaul to overhaul, coordinated operating system. Future revisions will be provided as necessary to support approved changes to the system [Ref. 15].

While the DOD instructions governing the development of an AIS treat the entire project as a single entity, the executors and managers of the TRIDENT LDS have chosen to break the overall project down into software subprojects (revisions) within the total project and to account for each revision by its own SDP milestone plan and its own time line. This revision phasing has been done in order to facilitate management of this vast and complex project and to accommodate progressive upgrades to TRIDENT Submarine System operational requirements.

2. Development Timing

DOD Directive 7920.1 establishes a policy which requires:

> "....As a goal, the overall AIS will be conceived and sized in a manner that will permit the development and evaluation of each module within 9 to 12 months after detailed design of the AIS has been completed. ....contribute to logic visibility, reliability, maintainability, and reduce the risk and cost associated with evaluation and validation."

The TRIDENT LDS is being developed using currently approved developmental concepts such as top down design, design walk-throughs, and chief programmer teams and has integrated

existing AIS capabilities but the complexity and uniqueness of the project does not support development within the DOD time frames. The TRIDENT LDS SDP indicates that the expected time required to progress from SDP Milestone 1 to SDP Milestone 2 – the system Definition/Design Phase – is 1 year and from SDP Milestone 2 to SDP Milestone 3 – the system Development Phase – 2 to 3 years [Ref. 15].

It must be noted that these time frames should be considered approximations of the time needed to complete these phases and are based on the projected size and complexity of the programs involved. Therefore, the dates indicated in Figure 6 are estimates and schedules to complete the various milestone phases should be based on the estimated length of time to complete each phase and the actual completion date of the preceeding milestone phase.

3. Development Documentation

Figure 9 shows a matrix containing AIS life cycle phases, SDP milestones, SDP contents (annexes), and system documentation applicable to each SDP milestone [Ref. 30]. The breakdown of system documentation by SDP milestones and AIS life cycle reflects decisions made by TRIDENT LDS managers. It should be noted that this matrix contains some departures from the guidelines promulgated by the SECNAV and DOD instructions.

a. The TRIDENT LDS has adopted a data systems development approach which begins with the preparation of a user's RS.

MILESTONES	0	I	II	III
Mission Element Need Statement, Annex A SDP ANNEXES	o Project Management Plan, Annex B with - B.1 System Description -- B.2 Management Plan -- B.3 Development Plan o Preliminary Acquisition Strategy, Annex C o Resources, Annex E, with — E.1 Cost/Benefit Analysis — E.2 Resource Requirements	o Acquisition Strategy, Annex C o Logistics and Training Plan, Annex D o Test and Evaluation Plan, Annex F o Updated Economic Analysis o Updated Resource Requirements o Updated Acquisition Strategy	o Updated Annexes	
SYSTEM DOCUMENTATION	o Requirements Statement	o Functional Description o Data Requirements Document o Hardware and Environmental Software Requirements	o Hardware Specifications o Subsystem Specifications o Data Base Specifications o Program Specifications o Operations Manuals o Users Manuals o Maintenance Manuals o Procurement Packages o Arrangement Drawings o Interconnection Drawings o Test and Analysis Reports o Program Trouble Reports o Test and Evaluation Plans - CDA Test Plans - User Acceptance Testing - Demonstrations - Readiness Review o Logistics and Training Plans - Maintenance Plans - Training Plans - Implementation Plans	o Program Change Requests o Test and Evaluation Plan - Post Installation Review - ADP Management Review - Effectiveness Reviews

Figure 9. TRIDENT LDS documentation relationships.

The TRIDENT LDS considers the RS a product of the Concept Development phase and has displaced the preparation of the Functional Description (FD) from this phase. Preparation of these two documents during the same development phase is incompatible although some overlap does occur because system designers often assist the users in refining and defining the problems to be solved. The RS, as explained in Chapter III A, represents the user's problem definition to be solved by the AIS and describes in terms of policy, concepts, objectives, and scope the requirements of the AIS. The FD builds from the RS and describes in detail the requirements of each system function identified in the RS including inputs, processing logic, files, and outputs. The FD is based on understanding and agreement between developers, users, and sponsors regarding the system's operational capabilities. The FD then is a "functional system design" document and acts as a transition vehicle from the RS to preparation of computer (hardware and software) design documents.

The development of an RS, while not specifically required by either DOD or Department of the Navy (DON) standards, is an important aspect of developing an AIS, especially a complex one such as the TRIDENT LDS. An RS supports a logical progression to creating both system and softward specifications and should be an integral step in the DOD life cycle phasing. An FD based on user agreement then is the next step to developing good specifications and logically is developed after the RS.

Failure to translate user requirements accurately and completely into both system and software specifications during the early stages of project development has been a major problem to the success of many software projects [Ref. 31]. Useful, quality specifications are very difficult and time consuming to creat [Ref. 32] and because of the level of effort and time constraints placed on the software project, there is a propensity for projects to develop and refine requirements as they are developed. These spontaneously generated requirements don't always accurately define the user's true needs and desires [Ref. 33] and can promote cost and schedule overruns. Additionally, poor requirements hence poor specifications can induce the following problems [Ref. 34]:

-- lack of definite guidelines for design personnel

-- difficulty in producing test plans and procedures because no set performance measurements have been established

-- user inputs are minimized because no clear statement of needs exists

b. The TRIDENT LDS has shifted the development of hardware, software and data base specifications into the Development Phase from the Definition/Design Phase. Again, this was done to accommodate the logical progression of the project and to facilitate building accurate specifications. The SECNAV and DOD instructions do not appear to support the

production of a long range, multifaceted, sequentially produced AIS and have a tendency of rushing through a system and crowding the functions together. This could result in more errors being produced than would be expected and more funds expended.

Requiring hardware, software, and data base specification to be developed when hardware and environmental systems have not been determined or developed is very difficult. If the hardware and environmental systems to be used are presently in production and will be either used as is or updated, then little or no problems exist for preparing these specifications. If, however, the hardware is still in a development and testing phase or only has had specifications drawn up on it, then the preparation of system specifications becomes much more difficult. Such is the case with the TRIDENT LDS project.

c. Developing a multifaceted AIS project creates another type of problem regarding scheduling and specifications. The TRIDENT LDS has six major functional areas to be developed and each of these functional areas has a number of modules or application operations (AO) internal to it. An FD is generally required for each AO [Ref. 35] but depending upon complexity, integration of project capabilities, on line timing requirements, and the like, an FD might only be necessary for each branch level within the project or possibly only at the total project level. The TRIDENT LDS project has 16 FDs developed/

to be developed and depending upon the revision the AO is in the target date for FD preparation and approval can vary by several years. The implementation of various AOs and LDS branches can and does have significant impacts on the operational characteristics of the entire system. Thus, when a project is faced with this type of situation, it can't wait to obtain all the FDs before progressing with software development or it may never satisfy the operational deficiencies addressed in the MENS within specified time constraints. Available FDs must be used to obtain projected hardware and environmental requirements and broad brush hardware, software, and data base specifications developed from these. Under these circumstances, it must be realized that specifications may require major revisions in the future as the equipment is brought closer to on line availability or as additional FDs are developed and approved.

4. Recommendation for Life Cycle Phasing

Figure 10 presents a possible realignment of AIS life cycle phases and SDP Milestones [Ref. 36]. Note that the Definition/Design Phase has been divided into two sections and an additional SDP milestone review and approval point added between the proposed Functional System Design Phase and the proposed Computer Design Phase. This allows a functional system design to be established, reviewed, and decided upon before progressing into the preparation of hardware, software, and data base specifications. During

SDS MILESTONES		0	I	II	III	IV
LIFE CYCLE PHASES	Project Initiation	System Policy and Concept	Functional System Design	Computer Design	Programming and Testing	Operation
SYSTEM DOCUMENTATION	Mission Element Needs Statement	Requirements Statement	Functional Description Hardware and Environment Requirements	Software Specs Data Base Specs Hardware Specs Test Bed Install	Program Specs Program Manual Operators Manual Maintenance Manual Users Manual Training Plan	

Figure 10. Realignment of SDP Milestones.

this phase the FDs would be prepared and agreed upon and initial hardware and environmental requirements established. Based on these approved parameters, the next stage of development then would create firm specifications upon which actual programming can be started. Additionally the Computer Design Phase would allow for the creation of a test bed system for in-house test and evaluation prior to on site deployment. Prior to progressing into the programming or development phase another SDP milestone decision point is encountered for additional project review and evaluation. This could be an important decision point when dealing with a long range innovative AIS.

The discussion of software life cycle phases and development of the TRIDENT LDS has established a foundation upon which to continue into Chapter IV. In the next chapter the budget process will be explored and, with the general knowledge gained in Chapter III as a basis, the budget categories of Design, Maintenance, and Management defined and applications of SDP Milestone cost and schedule variances provided.

IV. BUDGET GUIDANCE AND CONSTRAINTS

Life cycle management for Automated Information Systems (AIS) is a relatively new concept having been established late in 1978 by the Department of Defense (DOD) and applied to Department of the Navy (DON) AIS projects by the Secretary of the Navy (SECNAV) in late 1979. Little experience has been gained regarding AIS life cycle management. As more AIS developmental or revision projects are initiated under this concept, the more definition is required from it. AIS life cycle management is currently in a state of evolutionary change.

Chapter III pointed out that the TRIDENT LDS has added to the guidelines promulgated by the life cycle instructions and has modified the manner and sequencing by which a DOD software project is developed. This was done in an attempt to create a better base from which to build the system's computer programs and to smooth out and facilitate the development of this long range project.

This chapter will continue to delve into DOD's life cycle management program and will present a general discussion on budget policies, the SDP requirement to categorize TRIDENT LDS costs into Design, Maintenance, and Management categories, and the impact of the 15 percent cost and schedule variance on TRIDENT LDS budget formulation and cost accumulation.

A. BUDGET POLICY AND CONTROLS

SECNAV Instruction 5231.1A, DOD Directive 7920.1, DOD Instruction 7920.2, and the resultant Systems Decision Paper SDP) all provide some type of budget guidance to the development of the TRIDENT LDS. While budget guidelines and constraints are normal and can be expected in every fiscal situation, care must be exercised so that these guidelines are not too confusing, too lax, or too restrictive. If any one or a combination of these things occur, then the effectiveness and efficiency of the organization can be prejudiced. This section presents the rational behind budget policies and controls and shows how they can affect the operation of an organization. The subsequent sections of this chapter will demonstrate what has happened to the TRIDENT LDS because of budget policies and controls.

Budgeting is a management process which performs the following function [Ref. 37]:

- -- establishes the policy for an organization and sets its goals and objectives to attain that policy
- -- identifies weaknesses in an organization and provides a method through which they may be corrected
- -- controls and integrates diverse activities carried on by numerous subunits of a large organization
- -- provides a means of making an organization, agency, government, or individual accountable for its actions and through which performance may be judged

Appendix C lists some specific advantages and disadvantages to performing the budget function [Ref. 38].

Budgets are always created within a restricted financial environment [Ref. 39] and take strategic plans, policies, ideas, and decisions and breaks them down into specific operational level resources necessary to accomplish the assigned tasks. Every budget decision represents what someone wants to do or have someone else do [Ref. 40] and reflects the allocation of scarce resources to the alternatives which support the goals and objectives of the decision maker.

Because budgets are usually conceived in a top down fashion but prepared and submitted from the bottom up (always in the Federal Government), guidance and directions must be given to all levels and subunits within the organization on how to go about preparing the budget. This is done so that all the subunits will know what programs and activities will be emphasised or deemphasised during the upcoming budget period, what the estimated operating budget levels will be, what budget formats to use, when budgets are to be submitted for review and approval, who is responsible for preparing the budget submittals, what the criteria will be for evaluating the budget submissions, and any other general or specific instructions regarding the budget. Budget guidance is usually standardized and promulgated in official organizational bulletins, circulars, or operating procedures.

The standard budget guidelines are referenced and supplemented in the "budget call" for the specific budget period concerned. This budget call is the device which initiates the budget preparation and submittal phase and provides the budget guidance to management and operational levels.

Once the budget has been prepared, approved, and funds authorized, a budget execution system must be established. The budget exeution system provides directions to organizational subunits regarding actual budget operation and establishes a review plan by which to measure accomplishment of planned objectives [Ref. 41]. Much of the budget execution phase centers around budget limitations or budget constraints placed upon the obligation and expenditure of available funds. Budget limitations may be quite general or very specific and take form in the following ways:

-- restrict the amount of funds which may be obligated or expended over a specified length of time (usually the fiscal year or budget year or a portion thereof)

-- limit the programs, projects, or items on which funds may be expended and/or require higher authority approval before funds are expended in these areas

-- restrict the method through which funds may be expended - e.g., requiring higher authority approval before funds exceeding a certain amount per order may be expended or restricting the expenditure of funds to certain authorized individuals within the organization

-- requiring specific types of record keeping, accounting procedures, and reports to be generated and forwarded to higher authority for review

-- setting specific rates of expenditure in order to preclude running out of funds before the end of the budget period

-- establishing performance evaluation criteria by which the budget execution may be measured

Budget guidance and budget limitations are instituted with one or more of the following managerial ideas in mind: planning, coordinating, or control [Ref. 43;44].

Budget planning involves setting long range and short range plans for the entire organization and for each subunit within it [Ref. 43]. The organization's long range goals are brought down to short range objectives covering the budget period and then further subdivided down to the specific requirements for each subunit so that they will support attainment of the short range objectives and long range goals. If done correctly, budget guidance and controls will lead management at all levels to actively participate in and sincerely support planning for the organization's future. This in turn will tend to promote interest and enthusiasm toward the organization and its operations because middle and lower level managers will be able to see how their efforts go into the operation of the organization and how they can affect the overall scheme of things [Ref. 43].

Budget coordination refers to keeping all the organizational subunits working toward a common objective with regards to how each subunit affects the other subunits and the accomplishment of the stated objectives [Ref. 42]. For instance, the sales and production efforts of an organization have to be closely coordinated so that neither one adversely influences each other's operations and the objectives of the organization. If the sales department over commits the organization's production capability, resources may have to be reprogrammed into the production department in an attempt to catch up to the demand. If the demand can't be satisfied and customer dissatisfaction results, the organization's future sales potential may be compromised.

In an ADP development project, resources have to be coordinated and apportioned between the various modules and phases so that they support the timely, accurate development of the project. If testing is not resourced adequately, for example, the possibility exists that the system will not operate properly and will require the outlay of additional funds to correct it. Recovery time and cost to correct programming defects detected late in the development cycle will be much more expensive than the time and funds that would have been required to test properly the first time [Ref. 44]. Additionally, the customer may refuse to accept the project due to timing delays, failure to satisfy functional requirements, or significant cost overruns.

Coordination starts with a good integrated planning effort but relies upon the timely feedback to all managers of information relevant to correct operations and any revisions (additions, deletions, or changes) to original plans.

Finally, budget control concepts and devices stress financial accountability. They are geared toward making sure that no funds are used for other than approved purposes [Ref. 42]. Depending upon the severity of management's perceived need for budget control, the control devices employed may be so restrictive and limiting that middle level and low level management flexibility is impeded or so lax that fraud and waste is promoted. If too restrictive, managers spend too much time trying to stay within those fiscal and procedural requirements that they become unresponsive to emergent demands or changing environments. Workers and systems become so engrossed in staying within the constraints that their productivity decreases [Ref. 43]. If too lax or too confusing, budget controls may permit funds to be expended contrary to management's desires and the organization's goals subverted. Managers may also expend considerable amounts of time trying to determine exactly what is expected of them and then finding out that what they have done was not what higher authority actually wanted. Funds are wasted when this happens and a high degree of dissatisfaction created in the lower management eschelons.

B. DEFINING DESIGN, MAINTENANCE, AND MANAGEMENT BUDGET CATEGORIES

The Fleet Material Support Office (FMSO 96T) has been assigned the responsibility of being the TRIDENT LDS ADP Manager. One of the primary tasks assigned to FMSO 96T is the definition of and the budget preparation for the resources necessary to develop and maintain the TRIDENT LDS software project. One of the criteria for budgeting and cost acculu-lation which must be followed is the categorization of funds and costs into Design, Maintenance, and Management categories. These categories have been specified by the TRIDENT Submarine ILS Project Manager (NAVSEA PMS 396) by individual Work Breakdown Structure (WBS) numbers:

LDS - Management, B6J33C1A

LDS - Design, B6J33C1B

LDS - Maintenance, B6J33C1C

Funds designated for support of the TRIDENT Submarine Development Program are transmitted to FMSO via a Work Request (NAVCOMPT Form 140) citing these WBS numbers and the stipulation that funds cannot be exchanged between WBS numbers without the approval of the TRIDENT LDS Coordinator at Mechanicsburg, PA (SPCC 880).

Comparing these three WBS task descriptions with the AIS life cycle phases discussed in Chapter III, the WBS numbers tend to aggregate or consolidate a number of unique functions into broader categories and raise the questions of defining where design costs start and stop? what constitutes maintenance

costs? and what are management costs? in order to budget properly for them and avoid cost overruns.

1. Design

SECNAV Instruction 5230.6 entitled "Automatic data processing approval authority and acquisition/development threshold; delegation of" defines AIS development costs – and therefore those functions within life cycle phases that could be aggregated into the category 'design' – as:

> "... those expenditures which apply to the design, development, test, and implementation of the AIS. When determining the overall development cost to be compared to the AIS development threshold, sum the development costs from the time of approval of the Mission Element Needs Statement through the approval authority's acceptance of the system as operational (end of the System Development Phase). Development costs are one time (in-house and contractual) training, functional, personnel, ADP, and telecommunications costs. Do not include maintenance costs. ..."

While providing a time line for categorizing design costs and appearing to define them, this statement does not provide a clear enough description to differentiate between design and maintenance functions. If the WBS structure included a development category instead of a design category then possibly this definition could work. However, design more accurately describes a portion of the development functions and not the overall category.

Defining design costs (or development costs) as 'one time' costs seems to be overly restrictive. Software projects, especially large and complex ones, are usually produced over an iterative process of refining and redefining

design criteria in order to satisfy the RS. 'One time' can imply that design costs only should consider the first cut at developing the software projects but realistically it should include all costs up to and including the first time that the system satisfies the RS.

The same type of problem can apply to training costs. Should they be associated only with training systems' users and hardware/software operators or should such things as internal training of systems analysts and programmers assigned to the project be included in the costs.

Time lining design costs from the MENS approval through completion of the Development Phase also is questionable. Just because the system has gone operational doesn't automatically mean that all design functions have been completed [Ref. 23]. Operational commitments may have required expediting the on line capability before all the documentation had been prepared or waiving/postponing certain portions of the project. Completion of these items still belong under design requirements and should be costed as such [Ref. 23].

On the front end of this time line, approval of the MENS does not automatically mark the beginning of design/ development functions. A very complex project could require a significant amount of effort and time to produce a satisfactory problem statement, RS, or FD from which to proceed. According to the AIS life cycle phases described in the DOD

instructions, this type of effort falls into the Concept Development Phase. With the TRIDENT LDS project, FMSO's official functions formally start after approval of SDP Milestone 1 and continue on throughout the operational phase of the project. However, it does provide unscheduled technical assistance to the system users and sponsors in developing the RS. How then should this work be categorized? If it is considered design work, then it is tied to an SDP milestone and to a budget governed by a 15 percent variance allowance. But how can an accurate work load and budget be projected when work is performed on an as requested basis on an as yet undefined task? Logically this predesign work should not be included in the WBS design category but apportioned to either the maintenance category or the management category.

2. Maintenance

Approaching the separation of design and maintenance from the maintenance aspect also can produce an unsure situation. Computer software maintenance is generally associated to a system that has been operationally deployed [Ref. 31] and is responsible for correcting errors in the released product – corrective maintenance – or for providing minor alterations on the system – adaptive maintenance. Generally, software contractors are contractually obligated to perform software maintenance for the user/customer for a specified length of time.

A distinction is made between the types of product improvement because of the impact each has on the product's configuration management. Corrective maintenance or repair has little or no effect on the system's configuration status and is usually generated by detecting that the system does not perform the way it is supposed to perform because of improper coding, logic, or documentation. Adaptive maintenance on the other hand requires revision to system specifications, coding, and documentation and definitely changes the system's configuration account.

Adaptive maintenance is broken down into two categories — revisions and enhancements [Ref. 45]. Software revisions are changes to the product made necessary by a change in the system's environment, e.g., hardware changes or the addition/deletion of specific required transaction operations. Enhancements are not considered mandatory changes but merely improve the attributes or capabilities of the system. Enhancements allow the system to perform more operations thus making it attractive to a wider range of users.

Although commonly done, simply going operational with a system does not necessarily mark the end of system design development. McHenry and Walston [Ref. 23] warn against lumping revisions and enhancements into the maintenance category because of the redesign aspect common to both. Typically, they claim, software maintenance tasks are given to lower skilled persons and that very often when maintenance

on the system is requested by the user/customer, a redesign criteria is actually introduced by the request.

Software correction/change proposals are submitted by the user one at a time or in small groups. Scheduling the implementation of these requests should be determined by the criticality and risk involved with the change [Ref. 23]. The criticality or importance of a change request is often highly subjective and can be judged roughly by the delays and aberrations it produces in the system if not implemented promptly. By using the ploy of criticality, users can often get the software producer to process the change request quickly without thoroughly studying its scope. If the proposed activity alludes to a redesign of the software package and not simply minor corrections or minor tuning changes [Ref. 45], then this should be renegotiated with the user and a new RS obtained. At this point the system should reenter the design/development phase.

The following approach to separating design and maintenance has been taken by the TRIDENT LDS [Ref. 46]:

> "Design/Development includes all activity by the (TRIDENT LDS ADP Manager) from the approval of a system/application RS through initial implementation of the system. (This) activity includes the development of original documentation and application programs. The acquisition of hardware and environmental software necessary to support the new requirements is also considered part of the design/development process. Design/development does not include revisions to hardware, software and documentation that are required to support or modify the interfaces to existing systems. An existing system will become a design/development project if required revisions to the system are so extensive as to require the generation of a new requirements statement ..."

"Maintenance includes all activity by the (TRIDENT LDS ADP Manager) to enhance and/or modify an operational system. This includes the revision of documentation and application programs, the expansion/replacement of hardware and environmental software and facilities modifications, as required, to support the continued operation of the system, allow for normal growth in capacity and to correct inefficiencies and obsolenscence. Maintenance also includes the development and review/resolution of new requirements for future design development projects."

This method of determining whether a function and its related cost is in the Design or Maintenance category is supported by the information provided in this section. Approval of the system's RS provides a specific point at which time formal design work can commence and running the 'design line' out until acceptance of the system by the user accounts for all the iterations and changes necessary to bring that system to an operational status. Once the system has been accepted and all supporting documentation provided to the user, any work conducted on that system then becomes maintenance action. This definition also covers changing the scope of the system through maintenance requests. If the determination is made that the program changes or maintenance action requested by the user have the affect of changing the scope of the system, the system then reverts back to a design phase and a new RS renegotiated.

The life cycle management instructions do not assign nor call for SDP milestone approval for any functions occurring after SDP Milestone III (Operations/Development). Therefore, maintenance work does not fall under the cost and

schedule variance addressed in these directives. With this in mind, plus the parameters established by these definitions, budget estimates can be more accurately made for those functions and costs which are constrained by the budget variance.

3. Management

Management functions and management costs can be considered analogous to the function/costs of a service department or to overhead charges. A service department renders a service which contributes in an indirect manner to producing or providing a service but which itself does not directly participate in the process. Overhead is generally defined as indirect materials, indirect labor, overtime, supervision, fringe benefits or other expenses that can not conveniently be identified with or charged directly to a specific final cost objective [Ref. 47].

Unline direct material and direct labor, service departments and overhead are invisible parts of a final cost objective. Although they are invisible, these costs are a valid portion of the total costs and must be allocated back to the end product of the organization. This reallocation of costs is usually done on a predetermined rate (e.g., direct labor hours, lines of code written, machine hours) and is done to distribute these charges as equitably as possible.

Because of the requirements to report costs by Design, Maintenance, and Management categories, the TRIDENT LDS has approached the allocation of indirect charges from a

slightly different aspect. To facilitate cost accumulation into these categories, activities and applicable costs have been tied to either a line function or a staff function (see Figure 11). Line functions are those functions that can be tied directly to an LDS branch or Application Operation (AO) and include the personnel activities of the following FMSO departments: ADP Environmental Software Design (FMSO 94), Stock Point Systems Design (FMSO 95), UICP Systems Design (FMSO 96), and Financial Systems Design (FMSO 97). Additionally, while the Management Department (FMSO 92) is a staff department it performs work directly attributable to specific TRIDENT LDS functions and therefore has been included in the line department breakdown. These functions/costs are then designated either design or maintenance depending upon what LDS branch and AO the personnel are working on and the SDP milestone AIS life cycle phase that applies to that specific software project.

If the activity being performed does not originate from one of these departments or can be applied to many branches, then it is a staff function and classified as Management. Those TRIDENT LDS functions/costs which have been classified as Management are: TRIDENT LDS ADP Manager staff personnel (FMSO 96T), allocation of FMSO Comptroller Department (FMSO 91) activities for work performed for the TRIDENT LDS project, and allocation of the FMSO Operations Analysis Department (FMSO 93) for performance evaluation,

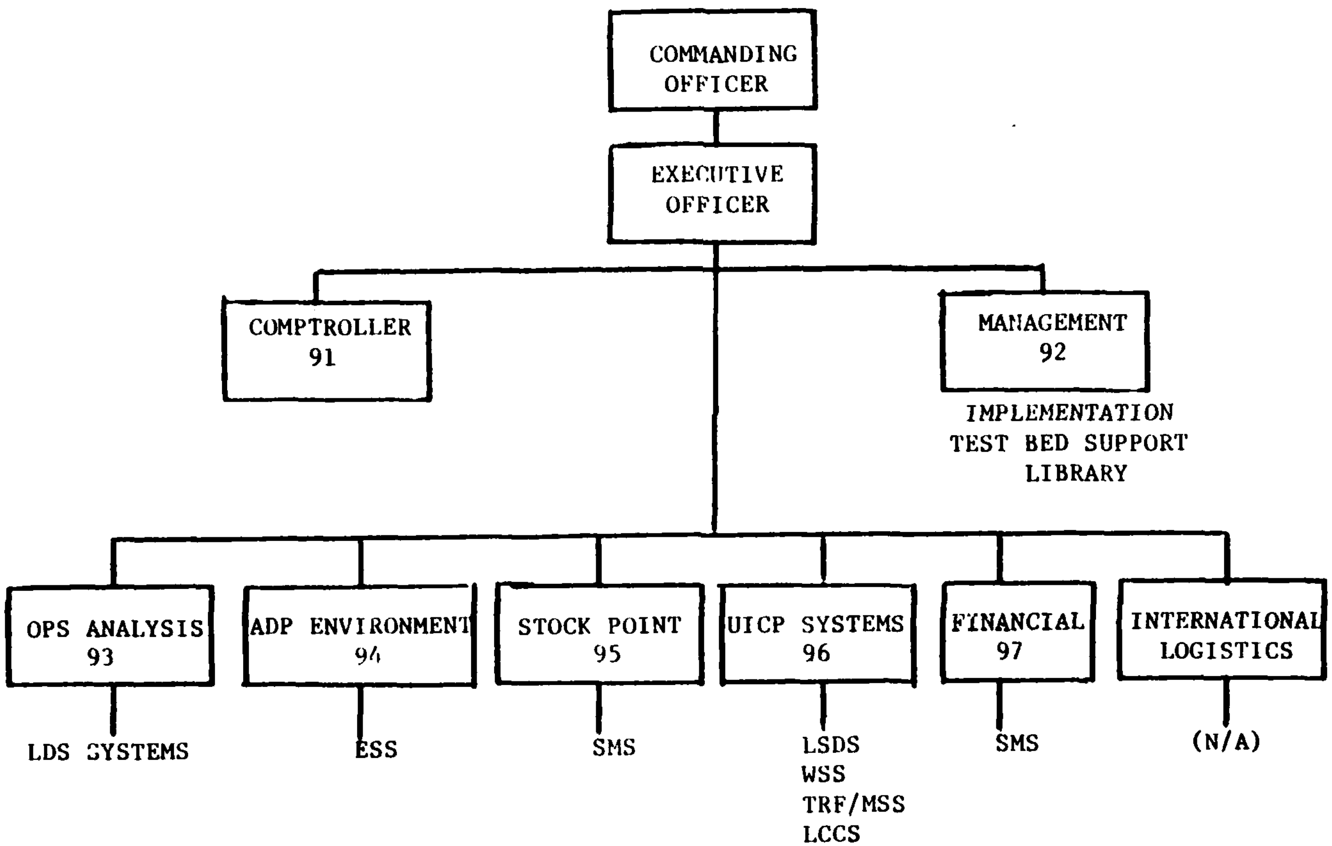

Figure 11. Fleet Material Support Office Organization Chart

modeling, simulation, and measurements taken against all the TRIDENT LDS programs written to project system capabilities. Additionally, material/supply costs for in-house TRIDENT LDS operations are collected in the Management category plus miscellaneous FMSO costs such as tuition, education, printing, and equipment rental/services if applicable to work performed in support of the TRIDENT LDS.

TRIDENT has chosen to exclude such costs as overtime, labor fringe benefits, and departmental supervisory costs from the Management category. For ease of categorization into Design, Maintenance, and Management, these costs are not treated as indirect or overhead costs but as direct costs to specific LDS functional branches or AOs. Within the budget, these costs are segregated out by job order number (e.g., LCCS branch management and administration costs and SMS branch management and administration costs) but then they are aggregated to either Design or Maintenance depending upon which stage of development the LDS branch/AO is in.

The approach to determining Management costs taken by the TRIDENT LDS is considered a satisfactory method. It facilitates the allocation of labor costs into the various categories by setting one criterion for determining whether the costs fall into the Design or Maintenance categories or into the Management category. If the function being performed (and its related costs) can be tied directly to producing a specific or a series of specific ADP products, then they are

classified as either Design or Maintenance as described earlier. All other labor costs then fall out into the Management category. If a further breakdown of these charges is requested, it can be acquired by selecting the appropriate job order number and collecting the costs charged to it.

Collecting all the miscellaneous costs and material/ supply costs into the Management category also makes the accumulation of costs easier although slightly less controllable. If these costs were allocated to each TRIDENT LDS branch or AO based on their usage then the Design and Maintenance costs would be more accurate. The cost to do this would very likely outweigh the benefit received, however, making the accumulation of these costs into the Management category more attractive.

C. SDP MILESTONE COST AND SCHEDULE VARIANCE

As explained in Chapter III, the DOD and SECNAV instructions have established specific decision points during the developmental phases of an AIS where the project is reviewed and assessed. This is done in order to periodically verify that its development continues to fulfill the customer's requirements and that it is doing so within projected cost and time constraints. These decision points are designated as SDP milestones and represent the major controlling steps to be attained in developing the AIS.

In addition to requiring these SDP milestone decision points, DOD and SECNAV have established parameters or constraints by which to evaluate the AIS project's efficiency in resource consumption. Specifically, the life cycle management program requires that a corrective action plan be generated, reviewed, and approved by the cognizant approval authority for the project if actual costs and time expended between SDP milestones exceeds the planning estimates by 15 percent or more. Although this variance constraint sounds relatively straightforward, it has been subject to a number of different interpretations regarding its meaning. The three most commonly occurring interpretations are as follows:

1. <u>Frozen Budget and Schedule Projections</u>

This interpretation of the cost and schedule variance constraint represents a literal translation of the instruction. That is, each milestone phase (Milestone 0 to Milestone 1, Milestone 1 to Milestone 2, and Milestone 2 to Milestone 3) stands by itself and is allowed up to but not including 15 percent slippage in either cost or schedule or both before notification and a corrective action plan is required. Further, once the initial cost and schedule estimates are made, they become "frozen" and remain plugged into the milestone matrix. These figures are then subject to the 15 percent variance allowance. Table III gives a simple example of this interpretation and Figure 12 shows the associated cost and schedule estimates curve and the variance curve. Note that

TABLE III

FROZEN BUDGET AND SCHEDULE PROJECTIONS

MILESTONE PHASE	0-1	1-2	2-3	TOTAL
ESTIMATES:				
COST	\$100	\$150	\$200	\$450
TIME	12 Mo.	12 Mo.	15 Mo.	39 Mo.
VARIANCE ALLOWED:				
COST	\$115	\$172.5	\$230	\$517.5
TIME	13.8 Mo.	13.8 Mo.	17.25 Mo.	44.85 Mo.
PERCENT CHANGE:	$\frac{517.5}{450} = 1.15$		$\frac{44.85}{39} = 1.15$	

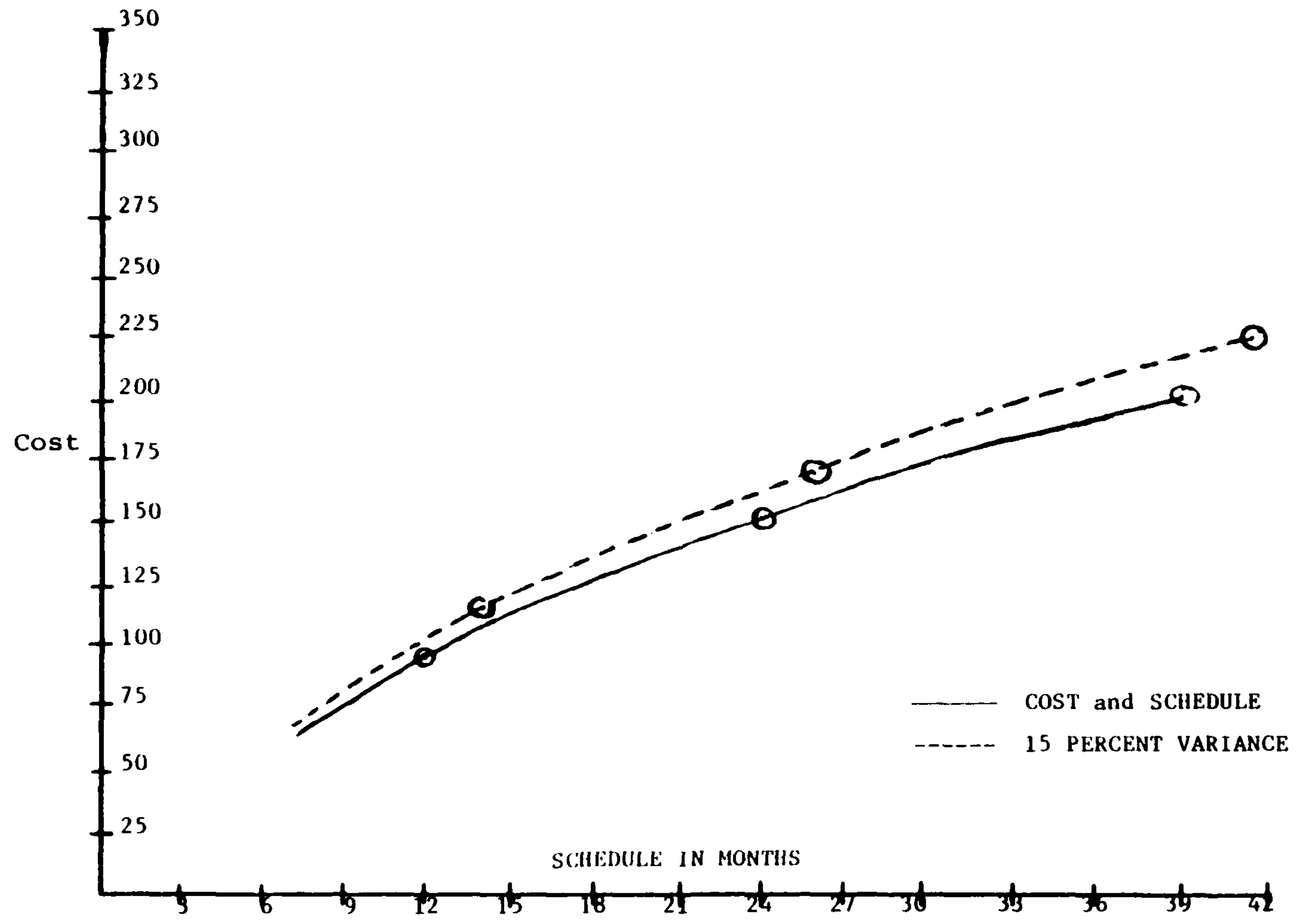

Figure 12. Frozen Budget and Schedule Projections

even if a milestone's actual costs and schedule vary from its estimated values, the succeeding milestones aren't affected and retain their original values. Schedule estimates for succeeding milestones simply begin when the previous milestone has been approved and the project allowed to continue. If this interpretation holds true, then an unjustified 15 percent variance for each milestone would only result in an overall unjustified variance for the project of 15 percent.

This interpretation and functioning of the SDP milestone variance constraints might be reasonable if the environment within which the AIS is being developed is known and stable. If the hardware and environmental systems to be used are currently operational (off the shelf procurement) and the customer's need (processing deficiency) is accurately defined and not an extremely complex task, budget and time schedules for the development of the AIS can be estimated very accurately. Archibald [Ref. 44] points out that the rate of expenditure of resources changes with each phase of AIS development, usually increasing with succeeding phases with a rapid leveling off or decrease near completion (see Figure 13). Developing a processing system from scratch using new ideas and new equipment increases the uncertainty associated with its creation. Thus, if the initial project goal is clear and concise, a more stable cost and schedule curve can be expected. A large initial area of uncertainty will result in greater awings in the cost and schedule curves

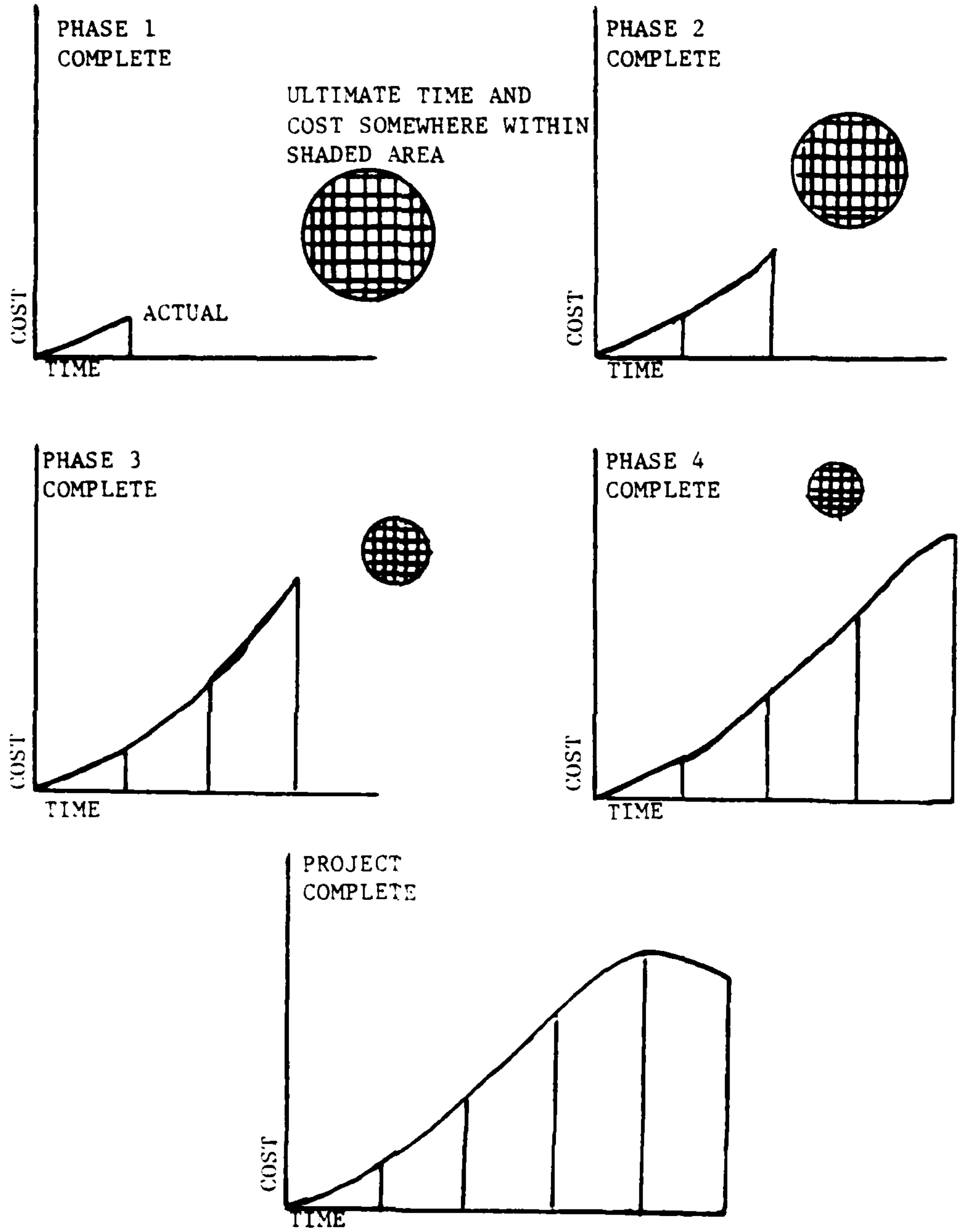

Figure 13. Relative Uncertainty of Ultimate Time and Cost by Phases

as more data is gathered and refined from each succeeding phase. This is the type of environment which leads to the next two interpretations of the SDP milestone cost and schedule variance.

2. Freezing the Active SDP Milestone Phase

This interpretation considers only the SDP milestone being worked on as being encumbered by the 15 percent cost and schedule variance. The active milestone becomes analogous to the current fiscal year and performance measurements made to evaluate its ability to meet the budget. The outyear milestone phases are treated as targets eligible for tuning and modifying as more data relative to the project becomes available. The outyear milestones become budget constrained once the milestone phase has been entered through approval of the preceeding milestone.

Table IV shows an example of what could happen if this enterpretation were allowed. The resultant cost and schedule estimates curve and its variance curve now appear stepped and can permit an actual cost and schedule variance greater than 15 percent (see Figure 14). This interpretation allows managers a great deal of flexibility regarding the development of the project and may result in a more complete and comprehensive capability from the end product. However, it can lead to sizable cost overruns and delay on-line availability of the system beyond reason. It also makes outyear budgeting and matching of anticipated revenues with expenditure requirements very difficult.

TABLE IV

FREEZING THE ACTIVE SDP MILESTONE

MILESTONE PHASE	0-1	1-2	2-3	TOTAL
ESTIMATES:				
COST	$100	$125	$200	$450
TIME	12 Mo.	12 Mo.	15 Mo.	39 Mo.
1ST PHASE:				
COST	$115	$150*	$230*	$495
TIME	13.8 Mo.	15 Mo.*	20 Mo.*	48.8 Mo.
2ND PHASE:				
COST	COMPLETED	$172.5	$275*	$562.5
TIME	COMPLETED	17.25 Mo.	24 Mo.*	55.05 Mo.
3RD PHASE				
COST	COMPLETED	COMPLETED	$316.25	$603.75
TIME	COMPLETED	COMPLETED	27.6 Mo.	58.65 Mo.
PERCENT CHANGE:	$\frac{603.75}{450} = 1.34$		$\frac{58.65}{39} = 1.50$	

* Subject to change without 15 percent limit

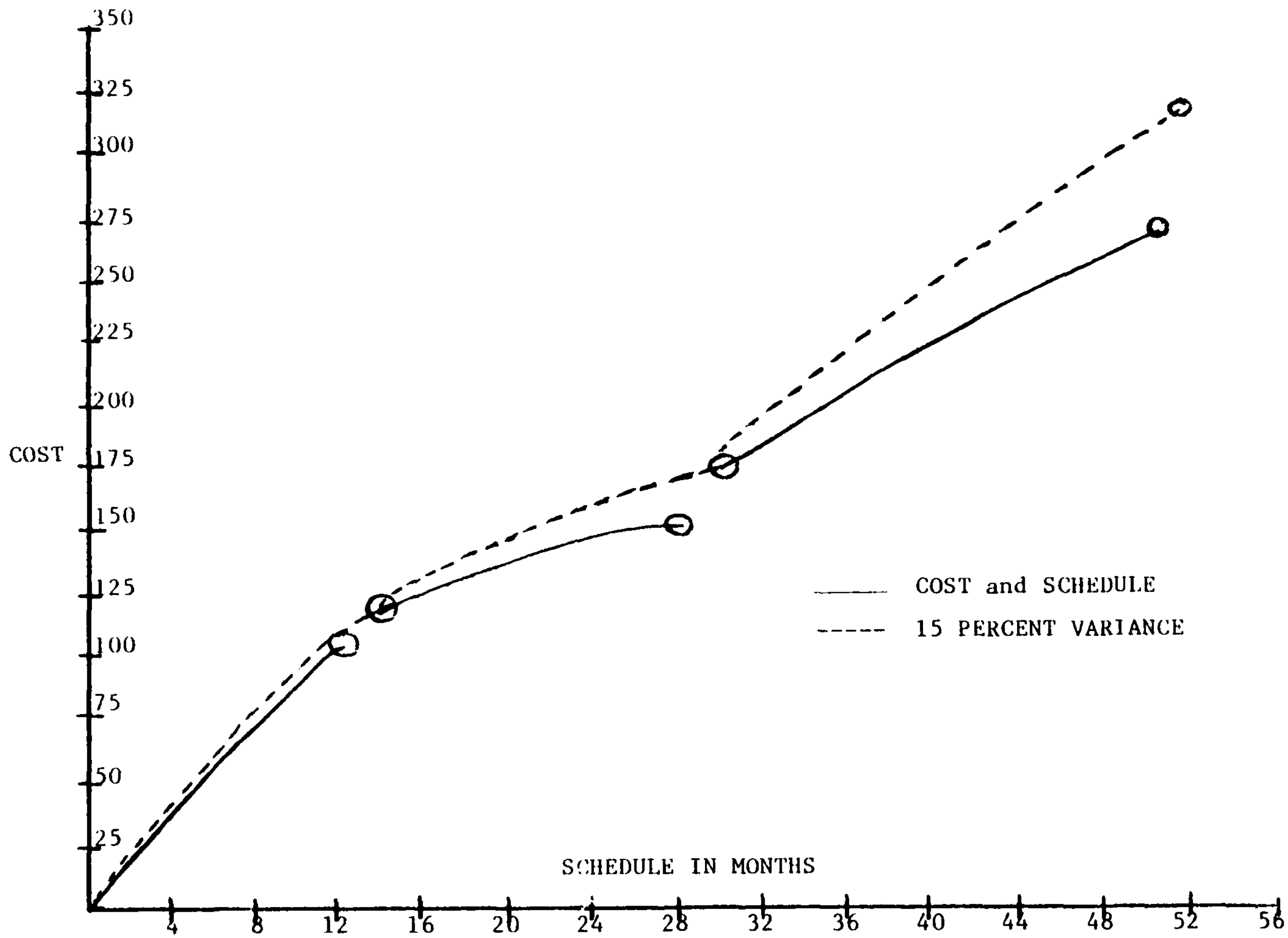

Figure 14. Frozen Active SDP Milestone Phase

3. Reprogramming Based on Missed Milestones

The final interpretation presented combines attributes of the preceeding two interpretations. As in the first presentation, the cost and schedule estimates are frozen. If a milestone can not be met within the 15 percent cost and schedule variance, the missed milestone must be justified and explained in detail and a corrective action plan established which will replan the remainder of the project. Based on the corrective action plan, the remaining milestones are reprogrammed both in cost and schedule. The milestones that are reprogrammed are not subject to the 15 percent variance during the reprogramming effort but once that milestone phase is entered, then the 15 percent variance constraint becomes binding. Reprogramming does not have to occur only at the point when the milestone is ready for review but can occur anytime within the milestone phase that it is realized that the cost and schedule estimates will not be met and that actual requirements will cause the project to exceed its variance limits. As in the second interpretation, this interpretation can result in free adjustments to the budget and schedule plans plus the 15 percent cost and schedule variance allowed by life cycle management. The estimated cost and schedule curve and the variance curve will appear stepped similar to thos presented in Figure 14.

D. TRIDENT LDS BUDGET AND SDP INTERFACE

DOD Instruction 7920.2 requires that an SDP be prepared following the approval of the MENS to facilitate and aid the decision making process regarding the continued development of the AIS project. It is the responsibility of the AIS project manager to create the SDP and maintain it in an updated status throughout the life cycle of the AIS. During the developmental phases of the AIS (SDP Milestones 1 through 3), the project manager is required to update and submit the SDP to the Office of the Secretary of Defense (OSD) at each succeeding SDP milestone.

As stated in Chapter III, the TRIDENT LDS project has been divided into five revisions over which the entire project capability will be implemented. Again, this phasing is being done to make it easier to manage this project and to coordinate project development with increased operational requirements of the TRIDENT Submarine System.

Because of this phased inplementation plan and the varying times at which the revisions' SDP milestones are scheduled to occur (see Figure 6, the decision was made by the TRIDENT LDS managers to update and submit the SDP annually for review and approval. The annual review will be supplemented with specific approval requests for each TRIDENT LDS revision to pass SDP milestones as required.

The annual submittal will serve to keep the OSD review process current with the TRIDENT LDS status and reduce the

number of times that the SDP would need to be submitted for a milestone review during a fiscal year. For instance, Figure 6 shows that three SDP reviews could occur in fiscal year 1981 - Revision 1 Milestone III (August 1981), Revision 2 Milestone II (September 1981), and Revision 3 Milestone I (January 1981). The annual SDP submission will allow a detailed look at the entire LDS project (all revisions and applicable LDS functional branches) at least once each year. The supplemental request will then bring the specific revision up to the SDP milestone review point.

The annual update will additionally be used to tie the SDP to the approved TRIDENT LDS budget. The budget is the best tool by which to enforce the cost and schedule variances allowed by AIS life cycle management. If the SDP is tied to the Program Objective Memorandum (POM), the POM is subject to more fluctuations and vagaries than is the budget. This could result in more mismatches and refiguring for the SDP.

Finally, an annual update and submittal will allow the most current cost and schedule estimates to be included in the SDP and actual cost and schedule usage displayed in the SDP and matched against the estimates.

V. SUMMARY AND CONCLUSIONS

A. SUMMARY

Initially, the reader was introduced to the need for better management and control of resources identified to the development of computer software projects. Excessive costs, schedule delays, and inability to satisfy customer requirements are common problems experienced with software development.

Implementation of Life Cycle Management for Automated Information Systems (AIS) demonstrated Department of Defense (DOD) concern for these problems and its attempt to mitigate their occurrence. A primary document in DOD's Life Cycle Management is the Systems Decision Paper (SDP) which contains all the essential information on the AIS and is created and maintained throughout the life of the AIS. Chapter II explains what the TRIDENT Logistics Data System (LDS) is, why its development is so important to the operations of the TRIDENT Submarine fleet, and discusses its transition to life cycle management and the SDP process.

Development of the TRIDENT LDS project was compared to the AIS developmental phases identified in the DOD life cycle management instructions in Chapter III. It was pointed out that the TRIDENT LDS is a long range, complex software system that has been divided into five revisions each of which correspond to increasing operational requirements for the TRIDENT

Submarine System. Total TRIDENT LDS system capabilities are to be phased in over the successive implementation of these revisions. Because of these factors, development of the TRIDENT LDS does not fit the mold of the DOD instructions and therefore has deviated somewhat from the life cycle phasing and software documentation sequences.

The categorization of TRIDENT LDS costs into Design, Maintenance, and Management was discussed in Chapter IV along with the 15 percent cost and schedule variances established by the life cycle management instructions. Breakdown of costs into these three categories and the interpretation of the cost and schedule variances tended to be subjective and not easily defined. Varying the way in which these guidelines are applied could result in very different budget requirements and cost breakdowns.

B. CONCLUSIONS

1. Ambiguous definition of customer/user processing needs coupled with precipitant development and design of system operating specifications can cause extensive rework of large software projects and result in drastic cost overruns and schedule delays. Specific definition and concurrence on the processing problem should be accomplished prior to performing any major software design work and prior to developing hardware, software, and program specifications, detailed functional and operational characteristics should be established. The

life cycle management instruction and DOD automated data systems documentation standards tend to compress and overlap the sequencing of software documentation preparation and do not specify the formulation of a detailed customer's Requirement Statement. This tends to create or foster problems when in fact the early planning stages of a software project should be based on alleviating problem areas.

2. The budget and cost accumulation guidelines provided by the life cycle management instructions and the SDP are subject to numerous interpretations which could cause confusion and errors in budgeting and could result in cost overruns. The criteria for placing costs into the categories of Design, Maintenance, and Management are satisfactory and facilitate cost accumulation.

Updating the TRIDENT SDP with the annual budget and displaying actual cost data against budgeted estimates and variance curves will provide a much more timely, useful management document. Creating new SDP cost and schedule baseline figures each year based on current information will present a more accurate status of the project but may run contrary to the expectations and policies of the approval agencies.

The author supports the decision to update and submit the SDP on an annual basis because this allows the most current data to be utilized in the decision making process and provides a firm budget figure by which to compare actual expenses.

Milestone phases which start and stop within the same fiscal year should be constrained by a 15 percent variance for that fiscal year. Milestone phases which start in one fiscal year but terminate in another fiscal year should not be constrained by one 15 percent variance curve for the entire time but should have a 15 percent variance curve for each fiscal year/portion of a fiscal year within which that milestone phase is active. Updates of the cost and schedule estimates in support of annual budget submissions should be allowed to migrate to the level supported by current data and not held constant to the initial estimates. As a management tool, justification should be provided for any estimate that exceeds the estimate provided in the previous year's POM process. The 15 percent cost and schedule variance is the recommended level at which time justification must be provided.

C. RECOMMENDATIONS

1. That AIS life cycle phasing be considered as suggested in Figure 10. This includes the definition and approval of a Requirements Statement and the addition of an SDP milestone review point that will assess development and approval of a functional system design prior to the development and approval of hardware and software specifications. This will enhance the probability of success for the project and reduce expensive rework of the project.

2. That the 15 percent cost and schedule variance described in DOD Directive 7920.1 and SECNAV Instruction 5231.1A be reviewed and clarification provided regarding its meaning and how it is to be applied to AIS budget formulation and execution.

3. That consideration be given to requiring the SDP to be submitted for review and approval at each major SDP milestone or at least annually if the time between milestones is greater than one year. This would help keep approval authority agencies more current with the progress of the software project and permit more timely feedback of actual cost and schedule performance.

APPENDIX A

INTEGRATED LOGISTIC SUPPORT (ILS) ELEMENTS

1. Maintainability and Reliability of equipment and components. Maintainability is the probability of restoring equipment to operating status within allowable time limits and reliability is the probability that the equipment will continue to function correctly for a specific period of time.
2. Maintenance Planning for organizational, intermediate, and depot level maintenance action.
3. Support and Test Equipment required by the operating forces and supporting maintenance activities.
4. Supply Support functions including provisioning, distribution and inventory replenishment of repair parts, spares, consumables and any other special supplies.
5. Transportation and Handling characteristics and requirements necessary to preserve, package, handle and transport all equipment and support items.
6. Technical Data including drawings; designs; operating manuals; maintenance instructions; inspection, test, and calibration procedures; and performance specifications.
7. Facilities needs based on operation and maintenance requirements including training requirements, test and evaluation functions, and installation and maintenance activities.

8. Personnel and Training requirements for operations and maintenance personnel and any training devices needed to support the program throughout its life cycle.
9. Logistic support funding for forecasting life cycle costs; planning and apportionment of required capital, operational and research and development costs; and allocation of available funds based on justified needs.
10. Management information and data for collecting, controlling and managing items 1 through 9.

APPENDIX B

TRIDENT LDS MAJOR BRANCH FUNCTIONS

1. Logistics Support Data System (LSDS). The LSDS branch is composed of TRIDENT unique program modules that support data acquisition, provisioning, support requirements, and planned maintenance requirements for the TRIDENT Submarine System. Data records will be established for each TRIDENT submarine equipment, component, and system requiring maintenance. Engineering and design data will be gathered along with all required maintenance actions and the logistic resources needed to support that maintenance. These records will be maintained and updated based on approved additions, deletions, and revisions. The LSDS branch has numerous TRIDENT unique programs which interface and allow data exchange between other standard Navy information systems. Maintenance requirements, test equipment, manpower skills, spare parts, and other data required to plan and schedule refit work at the TRIREFFAC will also be available.

2. Weapons Support System (WSS). The WSS branch consists of standard Navy programs that are currently operational on the computers at the logistics centers at Mechanicsburg, Pa. When the WSS interfaces with the appropriate TRIDENT LDS programs in the LSDS branch, the WSS will generate

standard Navy provisioning, Coordinated Shipboard Allowance List (COSAL), Incremental Stock Number Sequence List (ISNSL), and Load List products all tailored to the TRIDENT needs.

3. <u>TRIDENT Refit Facility Maintenance Support System (TRF/MSS)</u>. The TRF/MSS branch contains programs designed to provide automated planning, management, and support information to facilitate the performance of 18-day TRIDENT refits at the TRIREFFAC. The TRF/MSS will collect all planned maintenance, deferred maintenance, emergent maintenance, and other recurring maintenance requirements into refit work packages for the TRIREFFAC. Maintenance listings will be generated by separate work center, task, and system/equipment and will indicate all resources necessary to complete the required work. The status of each refit work package will be fed back into the TRF/MSS program branch after completion of the refit. This data will be used for updating the data files and creating the next refit work package for that submarine. Additionally, the TRF/MSS branch has program modules that will maintain and control a current inventory of technical data (drawings, publications, etc.) needed to support all refit maintenance activities; provide an automated tool crib system for the issue, receipt, location, and calibration status of tools and test equipment needed for refit work; monitor and schedule calibration requirements for tools and test

equipment on board each TRIDENT submarine; and finally, maintain inventory and maintenance records for TRIREFFAC Industrial Plant Equipment.

4. Logistics Change Control System (LCCS). LCCS is composed of programs which will plan and execute changes and alterations to equipment/components on board TRIDENT submarines. The completion of changes and alterations will be loaded back to the data files to ensure that the correct equipment/component configuration is reflected for proper logistic support.

5. Supply Management System (SMS). The SMS branch is a combination of existing and modified Navy programs and TRIDENT unique programs that will provide financial, inventory, and other supply functions to TRIDENT submarines and the TRIREFFAC. Capabilities will exist to monitor, follow up, and report requisition status and history; expedite material delivery; prepare and validate requisitions; provide automated receipt, storage, and issue of material at the TRIREFFAC; establish various cross reference files such as Job Order Number (JON) to requisition number files and stock number to manufacturers part number files; and generate financial reports required by the Navy Comptroller (NAVCOMPT) and processing reports required by the Fleet Accounting and Disbursing Center (FAADC).

AD-A118 197 NAVAL POSTGRADUATE SCHOOL MONTEREY CA F/G 5/1
TRIDENT SUBMARINE LOGISTICS DATA SYSTEM (LDS): A CASE STUDY IN --ETC(U)
MAR 82 G W HIZA

UNCLASSIFIED NL

2 OF 2

END
DATE
FILMED
9-82
DTIC

6. Environmental Software Systems (ESS). ESS provides the operational environment to support, control, and coordinate TRIDENT LDS programs operated on TRIDENT LDS hardware.

APPENDIX C

ADVANTAGES/DISADVANTAGES OF BUDGETING

A. ADVANTAGES

1. It forces early consideration of basic policies.
2. It requires adequate and proper organization and assignment of responsibility.
3. It compels all members of management from the top down to participate in the establishment of goals.
4. It forces management to put down in figures what is necessary for satisfactory results.
5. It compels all members of departmental management to make plans in harmony with plans of other departments.
6. It compels management to demand adequate historical accounting data.
7. It instills into all levels of management the habit of timely, careful, and adequate consideration of all factors before reaching important decisions.
8. It compels management to plan for the most economical use of labor, material, facilities, and capital.
9. It pinpoints efficiency or its lack.
10. It promotes understanding by management of their co-workers' problems.
11. It forces a periodic self-analysis of the organization.

12. It checks progress or lack of progress toward the objectives.
13. It forces management to give timely and adequate attention to the effect of the trend in general environmental conditions.
14. It promotes knowledge at lower levels of basic policies and objectives.

B. DISADVANTAGES

1. The budget plan is based on estimates. The estimates must be based on all available facts and good judgment in interpreting and using the results.
2. Budgetary programs must be continually adapted to fit changing circumstances. It can not be installed and perfected in a short time. Budget techniques must be continually adapted and new techniques tried with changing situations. Development may take several years and management has to remain patient with it.
3. Execution of the budget will not occur automatically. Responsible managers must get behind it and continuously press for its accomplishment. All levels of management must be sold on budgeting and participate in it.
4. The budget will not take the place of management and administration. It is a tool of management and must be used as such. A budget must be treated as a servant

and not as a master; it should not be assumed to be perfect and impossible to change.

LIST OF REFERENCES

1. U.S. Comptroller General Report FGMSD-80-38, Wider Use of Better Computer Software Technology Can Improve Management Control and Reduce Costs, p. 1, April 29, 1980.

2. U.S. Comptroller General Report FGMSD-77-34, Millions in Savings Possible in Converting Programs from One Computer to Another, p. 1, September 15, 1977.

3. Harris, J.N., Complexity as a Factor of Quality and Cost in Large Scale Software Development, MS Thesis, p. 11, Naval Postgraduate School, 1979.

4. U.S. Comptroller General Report FGMSD-80-4, Contracting for Computer Software Development - Serious Problems Require Management Attention to Avoid Wasting Additional Millions, p. 9-10, November 9, 1979.

5. Prokop, Jan, Computers in the Navy, "Governing Regulations", p. 25-34, Navy Institute Press, 1976.

6. TRIDENT System Project Naval Material Command Draft Logistics Data Systems System Decision Paper, p. A1-1, unpublished.

7. Ibid., p. A1-4.

8. Chief of Naval Material Command Instruction 4000.20B, "Integrated Logistic Support Planning Policy", p. 2, 27 June 1975.

9. Ibid., p. 3.

10. Department of the Navy, TRIDENT Systems Project, TRIDENT System Integrated Logistics Support Plan, PM2/42 4000 Ser 942, 29 November 1979.

11. TRIDENT System Project Naval Material Command Draft Logistics Data Systems System Decision Paper, p. E1-4, unpublished.

12. Ibid., p. A2-1 - A2-18.

13. Ibid., p. B1-5.

14. Ibid., p. B3-17.

15. Ibid., p. B3-16.

16. Ibid., p. B2-4.

17. Metzger, Philip W., Managing a Programming Project, p. 5, Prentice Hall Inc., 1973.

18. Cortada, James W., Finance, Budgets, and Cost Control in Data Processing, p. 56-67, Prentice Hall Inc., 1979.

19. Lucas, Henry C., Information Systems Concepts for Management, p. 220-221, 332-334, McGraw-Hill, 1978.

20. Gunther, Richard C., Managment Methodology for Software Product Engineering, p. 17, Wiley, 1978.

21. Lucus, Henry C., Information Systems Concepts for Management, p. 241, McGraw-Hill, 1978.

22. Ibid., p. 243-247.

23. McHenry, Robert C. and Walston, Claude E., "Software Life Cycle Management: Weapons Process Development", IEEE Transactions on Saftware Engineering, Vol. SE-4, No. 4, July 1978.

24. Metzger, Philip W., Managing a Programming Project, p. 42-45, 165-166, Prentice Hall Inc., 1973.

25. Lucas, Henry C., Information Systems Concepts for Management, p. 253, McGraw-Hill, 1978.

26. Metzger, Philip W., Managing a Programming Project, p. 63, Prentice Hall Inc., 1973.

27. Ibid., p. 46.

28. Ibid., p. 113-121.

29. Lucas, Henry C., Information Systems Concepts for Management, p. 263-264, McGraw-Hill, 1978.

30. TRIDENT System Project Naval Material Command Draft Logistics Data Systems System Decision Paper, p. B2-23, unpublished.

31. Cave, William C. and Salisbury, Alan B., "Controlling the Software Life-Cycle – The Project Management Task", IEEE Transactions on Software Engineering, Vol. SE-4, No. 4, July 1978.

32. Yeh, R.T. (Ed), Current Trends in Programming Methodology, Vol. I, Prentice Hall Inc., 1977.

33. Harris, J.N., Complexity as a Factor of Quality and Cost in Large Scale Software Development, MS Thesis, Naval Postgraduate School, 1979.

34. Ibid., p. 37.

35. Navy Fleet Material Support Office Instruction 5230.20, "CDA Development Handbook", p. 3-15, 28 September 1979.

36. Interview with LCDR R. T. Johnson, TRIDENT LDS ADP Manager on 11 February 1981.

37. Lee, Robert D. and Johnson, Ronald W., Public Budgeting Systems, p. 13-22, University Park Press, 1973.

38. Welsch, Glenn A., Budgeting: Profit-Planning and Control, p. 12-13, Prentice Hall Inc., 1957.

39. Lynch, Thomas D., Public Budgeting in America, p. 2, Prentice Hall Inc., 1979.

40. Ibid., p. 5.

41. Ibid., p. 144-146.

42. Ibid., p. 9-10.

43. Welsch, Glenn A., Budgeting: Profit-Planning and Control, p. 6-11, Prentice Hall Inc., 1957.

44. Archibald, Russell D., Managing High Technology Programs and Projects, p. 23-25, Wiley, 1976.

45. Gunter, Richard C., Management Methodology for Software Product Engineering, p. 170, Wiley, 1978.

46. Ships Parts Control Center Memorandum for the Record 881B/C8H 5230 Subject: TRIDENT LDS Financial Management and Revision Boundaries, February 5, 1981.

47. Matz, Adolph and Usry, Milton F., Cost Accounting Planning and Control, p. 251-253, South Western Publishing Co., 1976.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, California 93940	2
3.	Department Chairman, Code 54(JS) Department of Administrative Sciences Naval Postgraduate School Monterey, California 93940	1
4.	LT G. W. Hiza, SC, USN 375 Hummingbird Lane Livermore, California 94550	1
5.	Dr. Shu Liao, Code 54(LC) Department of Administrative Sciences Naval Postgraduate School Monterey, California 93940	1
6.	Dr. N. R. Lyons, Code 54(LB) Department of Administrative Sciences Naval Postgraduate School Monterey, California 93940	1
7.	LCDR R. T. Johnson, SC, USN Navy Fleet Material Support Office P. O. Box 2010 Mechanicsburg, Pennsylvania 17055	2

DATE

FILMED

8